izdaje bigz
izlazi jedanput mesečno

računari

38
maj 1988
cena 1500 din
YU ISSN 0352 7271

časopis za popularizaciju informatike i računarstva



komercijalni softver

smalltalk/v

tehnika programiranja

borci u memoriji

umetak na 32 strane

mala pc biblioteka **lotus 1 2 3**

serije

kućni roboti • stono izdavaštvo

računari
maj 1988
cena 1.500.—
Izdaje BIGZ
časopis za popularizaciju informatike i računarstva
Izlazi jedanput mesečno
YUISSN 0352-7271
Izdaje
Beogradski izdavačko-grafički zavod
11000 Beograd
Bulevar vojvode Mišića 17
•
Generalni direktor
Dobrosav Petrović
Glavni i odgovorni urednik časopisa „Galaksija"
Stanko Stojiljković
•
Glavni i odgovorni urednik
Jovan Regasek
Stručna redakcija
Žarko Berberski Voja Gašić
Slobodan Perović Dejan Ristanović
Jovan Skuljan prof dr Dušan Slavić
Nevenka Spalević Zoran Životić
Anđelko Zgorelec
Sekretar redakcije
Zorka Simović
Likovno grafičko uređenje
Mirko Popov
Marketing
Sergije Marčenko
Stalni saradnici
Neda Aleksić Slobodanka Ast Žarko Berberski Viktor Cerovski Zoran Cvijetić Nincelav Ćalalić Voja Gašić Željko Jurić Blažimir Miše Zoran Obradović Slobodan Perović Miodrag Potkonjak Aleksandar Radovanović Dejan Ristanović Jelena Rupnik Duško Savić Dušan Slavić Jovan Skuljan Nevenka Spalević Vlada Stojiljković Saša Svilica, Milan Tadić Žarko Vukosavljević Anđelko Zgorelec Zoran Životić
Izdavački savet „Galaksije"
Dr Rudi Dekleva prof dr Branislav Dimitrijević (predsednik) Radovan Draškovič, Tanasije Gavranović Živorad Gligić Esad Jakupović Velizar Maslać Nikola Pajić Željko Perunović prof dr Momčilo Ristić Vlada Ristić dr inž. Milorad Teofilović Videjko Veličković, Velimir Vasović Milivoje Vuković
•
Adresa redakcije
11000 Beograd
Bulevar vojvode Mišića 17/III
Telefoni
650-161 (sekretarijat)
653-749 (redakcija)
650-528 (prodaja)
651-793 (propaganda)
Rukopisi se ne vraćaju
Štampa
Beogradski izdavačko-grafički zavod
11000 Beograd
Bulevar vojvode Mišića 17
•
Pretplata
Jugoslavija
Za jednu godinu — 18 000 —
Za šest meseci — 9 000 —
Na žiro račun RO BIGZ 60802-603-23264
Inostranstvo
Za inostranstvo 25 USD 41 DDM 18 Latg 34 CHF 147 SEK 13 GBP 141 FRF 292 ATS ili 36 000 din
Na žiro račun RO BIGZ 60811-620-16101-620701-999-03377
Na osnovu mišljenja Republičkog sekretarijata za kulturu broj 413 77/72-03 i „Službenog glasnika" broj 26/72 ovo izdanje oslobođeno je poreza na promet

Sadržaj



„Devojka i računar su sve što je hakeru potrebno" — Bata Data. Sve to za vas je okupio Bratislav Đorđević ispred objektiva Darka Todića. Možda hakeri kod kuće imaju i bolje mašine, ali zato manekenka Jasna Đorđević i učenica Ksenija Kopanja mogu da izdrže mnogo oštriju konkurenciju

Nonsense in BASIC

Ćar i vajda

Ja sam najobičniji frazer i demagog nemam blage veze sa životom za pisanje sam trube a u računarstvo se razumem kao magarac u kantar. Pismo s takvim karakteristikama stiglo mi je preko Računara. Čitalac koji ga je poslao potpisao se kao Zoran Mudrinić apsolvent arhitekture ali na bih stavio ruku u vatru da mu je to pravo ime

U prošlom broju objavljen je moj komentar Cena pameti. Pišući taj tekst mislio sam da uzimam u zaštitu programere. Zoran je shvatio da napadam pirate i to je izazvalo njegov gnev. Šta ja uopšte imam protiv ljudi koji pošteno rade i zarađuju nikome ne sole pamet (kao ja) i ne prodaju zjala na novinskim stupcima (opet kao ja) nego dobre programe? Da nije pirata pitanje je kako bi se u nas razvijalo računarstvo. Prosečan Jugosloven bi malo sutra mogao da kupi program po ceni koja važi u inostranstvu

Kad se sve sabere bar tako tvrdi Zoran od pirata svi imaju koristi

a) Vlasnici računara jer za male pare dolaze do skupih programa

b) Računarstvo uopšte jer bez softvera nema razvoja informatičke pismenosti

c) Računarski časopisi jer uzimaju novac za oglase

d) Sami pirati jer im prihodi od preprodaje programa dopunjuju lični budžet (Zoran kaže da bi iz materijalnih razloga odavno napustio studije a da to nije učinio zahvaljujući piratovanju)

Nemam nameru da polemišem. Ako se izuzmu uvrede upućene meni možda je Zoran zaista u pravu. Spreman sam čak i da prihvatim njegove stavove ali pod jednim uslovom. Neka on uradi projekt za porodičnu kuću. Ja ću taj projekt fotokopirati i prodavati preko oglasa po hiljadu dinara primerak. Bagatela! Od toga će svi imati koristi

a) Graditelji kuća jer će za male pare doći do skupog projekta

b) Stambena izgradnja uopšte jer će veći broj stanova ublažiti stambenu krizu

c) Novine jer će uzimati novac za oglase

d) Lično ja jer na drugi način ne mogu da skupim novac za stručno usavršavanje u inostranstvu

e) Zoran i to istu onakvu vajdu kakvu ima programer čiji program on preprodaje

Bata Bajt



Svet

Programari na cani

Američka softverske kuće sve češće sklapaju aranžmane sa evropskim programarima koji uživaju visoku reputaciju — *Microprose* je, na primer, nedavno dao nekoliko oglasa u okviru kojih se traže vrhunski programeri koji će raditi na vrhunskim projektima

Proširenja

Ozbiljni džojstik

Iako reč *joystick* obično prevodimo kao „palica za igre", džojstik može da bude i te kako ozbiljna naprava. Američka firma *CH Products* ((619)744-8546) za osamdesetak dolara prodaja izuzetno kvalitetno urađene analogne džojstika sa drajverima za brojne PC i „epl 2" programe za crtanje i dizajn. Za svaki se slučaj dodaje da je džojstik savršena kontrolna palica za sve *Flight Simulator-e*



Proširenja

Portabl uz portabl

Mnogim vlasnicima portabl PC računara mnogo nedostaje numerička tastatura koja je obično žrtvovana da bi kompjuter stao u tašnu. Rešenje je kupovina portabl numerička tastatura firme Genest Technologies ((800)826--9641) koja se sa PC-jem povazuje uz pomoć pola metra dugačkog kabla — tastaturu nosite u nekom džepu torbe a onda je po potrabi povežete sa COM1 ili COM2 portom. Cena ne zvuči strašno — 130 dolara.



Profi oprema

Motorola i personalci

Nije se samo Intel pretplatio na 32--bitne mašine. Omega je nova radna stanica britanske firme Lloyd I/O Inc (telefon (800)227-3719) zasnovana na Motorolinom mikroprocesoru 68020 i aritmetičkom koprocesoru 68881 koji rade na 12.5—25 MHz, opremljena megabajtom RAM-a koji se gotovo neograničeno proširuje, grafikom 640×480 u 16 boja, hard diskom od 20—100 megabajta i novim multiprogramskim operativnim sistemom OS9/68 K. Cene se kreću od 4750 (12.5 MHz) do 6895 (25 MHz) dolara

Komercijalni softver

Pritajani statistički kalkulator

Pisanje memorijski rezidentnih programa za PC koji simuliraju razna kalkulatora je očito ušlo u modu. Posle programerskog HP-16C i „starog dobrog" HP-41C firma *Cisar lake Research* ((713) 523-7842) je ponudila statistički kalkulator koji obezbeđuje sedam standardnih, 12 statističkih i brojne memorijske funkcije. Vredi pomenuti srednju vrednost, standardnu devijaciju, standardnu grešku, linearnu regresiju, korelaciju, verovatnoću

Cena je više nego ohrabrujuća — 22 dolara

Proširenja

NEC-ovo stono izdavaštvo

Ako je mašina koju koristite za tehničku pripremu knjiga neki IBM PC, zanimaće vas NEC-ova MonoGraph kartica koja obezbeđuje prikaz kompletne A4 stranice teksta na ekranu. Hardver grafičke kartice zasniva se na Intelovom grafičkom kontroleru 82786 koji, u saradnji sa izvanrednim 20-inčnim monitorom, obezbeđuje do skora neverovatnu grafičku rezoluciju od 1024×1024 tačaka. Što se softvera tiče, u cenu od 1999 dolara uračunati su i drajveri za *Venturu, PageMaker, MS Windows* i *Lotus*

Ako pozovete (312)860-9500 saznaćete još jednu dobru osobinu MonoGraph kartice — tvrdi se da je 100% CGA kompatibilna!

Komercijalni softver

Zakrpa *dBASE-A*

Jedna od velikih mana dBASE-a je nemogućnost direktnog povezivanja sadržaja nekoliko datoteka ili prenošenje dela slogova iz jedne datoteke u drugu — može se doduše napisati programčić koji rešava problem ali ništa ne može da zameni stalno pristupačnu opciju menija! Zato je tu firma Proximity Technology ((305)566-3511) koja za 100 dolara prodaje Friendly Finder, program koji uz sve izloženo omogućava prenošenje podatka iz dBASE-a u aplikaciju koju trenutno koristite. Dani kada ste da biste u poslovno pismo uneli nečiji telefon napuštali WordPerfect, startovali dBASE, pronalazili slog, prepisivali telefon na papir, napuštali dBASE, startovali WordPerfect, učitavali dokument i prekucavali broj dakle pripadaju ružnoj prošlosti



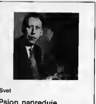
Komercijalni softver

Animacija na „amigi"

Sculpt 3-D i Animate 3-D su novi programi za „amigu" koji obezbeđuju modeliranje, crtanje i animaciju u realnom vremenu, nudeći kontrolu svih 4096 „amiginih" boja i biblioteku oblika koja obuhvata lopte, kupe, prizme, alate i sve standardne arhitektonske projekcije trodimenzionalnih objekata. Pošto je crtanje završeno, možemo da pomeramo tačku iz koje posmatramo objekat i tako ostvarimo utisak približavanja, udaljavanja ili zaokretanja. Obezbeđeno je, jasno, i pomeranje simuliranog izvora svetlosti i, prema tome, rad sa senkama i neprovidnim objektima.

Sculpt 3-D košta 100 a Animate 3-D 150 dolara, dok je telefon firme Byte by Byte (512)343-4357.





Svet

Psion napreduje

Mali Psion već vredi gotovo 17 miliona funti! Ogromni uspeh Psion Organizer-a II džepnog računara koji je čitaocima „Računara" predstavljen pre dva meseca omogućio je Psion-u i njegovom vlasniku Dejvidu Poteru (David Potter) da pokušaju prodor i na druga tržišta naročito američko i nemačko

Mikroprocesori

ARM Kartica za PC-ja

Još od pojave Acornovog „arhimeda" pričalo se o ARM kartici za PC-je Obećanje je (što nije uvek bila Acornova praksa) brzo ispunjeno — vlasnici PC-ja koji žele da potroše 1 000 dolara da bi svoj „najbolji personalac na svetu" pretvorili u „najbrži personalac na svetu" treba samo da nabave *Springer Board* Kartica sadrži 32-bitni ARM procesor 4 megabajta RAM-a i odgovarajuće interfejse koji obezbeđuju kartici saradnju ali i samostalnost — RISC procesoru možete da naredite da radi neki komplikovani proračun dok se vi zabavljate vernim Lotusom Nije naravno, isključena ni obrnuta varijanta

Acorn uz *SpringerBoard* isporučuje C, fortran paskal, lisp i bejzik kompajlere i tvrdi da proširenje obezbeđuje brzinu od 4 MIPS-a

Svet

Skok cena

Iako smo navikli da cene računarske opreme stalno padaju povremena nestašice čipova mogu da izazovu i obrnute efekte — setimo se samo statičkih memorije 6116 za „galaksiju" Sada su na ceni dinamički RAM-ovi — porast cena naterao je firme Amstrad i Atari da povise cena svojih modela za desetak procenata Povećanje cene međutim *de*luje sa zadrškom koja ne škodi prodaji da ste „amstrad 1512" ili „atari ST" nabavili pre prvog aprila platili biste ga po staroj ceni!

Nestašica čipova je svakako privremena, jer je na tržištu sve više interesantnih RAM-ova NEC i Motorola su, na primer, konačno najavili masovnu proizvodnju dinamičkih RAM-ova od jednog megabita

321

Profi oprema

Unix na kartici

Torch vam nudi priliku da proverite poznavanje juniksa a koje ste stekli čitajući našu malu seriju napisa koja se u ovim „Računarima" završava — pozovite (britanski) broj 0223-841000 uplatite 4,000 funti (svaka se nauka plaća) i očekujte VME32QX karticu koja zapravo predstavlja kompletnu Unix procesorsku ploču zasnovanu na procesoru 68020 (20 MHz) koprocesoru 68881, specijalnom MMU i DMA kontroleru zasnovanom na RISC arhitekturi i VME magistrali Potreban vam je još samo neki terminal koji može da podrži rezoluciju 1024·768 i spremni ste za Login dijalog

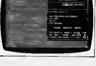


Komercijalni softver

Rezidentno uputstvo

Usred pisanja programa mozak vam je stao i nikako ne možete da se setite sintakse neke naredbe ili strukture Pružate levu ruku otvarate fioku tražite uputstvo prevrćete listove ili pritisnete Alt CTRL i još nešto — tekuća aplikacija se prekida i ulazite u rezidentni priručnik firme Santa Rita Software ((214)727-9217) Šetate se kroz menije i začas pronalazite željenu informaciju o DOS komandi bejzik Turbo C Pascal Prolog Modula 2 ili C instrukciji!

Pre svega toga treba samo platiti pedesetak dolara prepisati program na hard disk i još jednom editovati sirotu datoteku AUTOEXEC BAT





Štampači

Super trake

Svi znamo koliki problem predstavljaju trake za štampače — treba ih nabavljati iz inostranstva dugo stajanje im ne prija brzo blede Najlonske trake firma SBS Computer Supplies ((273)726-331) prilagođena Amstradovom štampaču 9512 su najavljena kao prava mala revolucija tvrdi se da će trajati bar četiri puta duže od karbonskih a da cena neće preći sasvim konkurantnih 4 25 funti Da vidimo i to čudo!

Igre

Avanture za „plus 3"

Kakve korist od diska ako nema igara koje će zahvaljujući njegovim kvalitetima biti beskonačno duge i teške? Vlasnici „spektruma plus 3" koji obožavaju avantu*ra* mogu da se obrate firmi *Topologika* ((733)244-682) i nabave *Kingdom of Hamil Countdown to Doom 2 Philosopher s Quest 2* i *Acetron.* Sa srećom — trebaće vam!

Masovna memorija

AP programi na traci

Ne nismo mislili da su vam diskete i hard diskovi za AT-a dosadili i da se vraćate kasetofonu ATC-8 strimer traka firme *Catamount* (američki telefon ((805)584-2233) je izvanredan medij za skladištenje velikih količine podatake i *backup* ali nudi i jednu stvar koja mnogo nedostaje konkurentskim strimerima kompatiblinost sa 9-trečnim industrijskim standardom O blagodetima ove kompatiblincsti vredi govoriti samo nekome ko nije radio u iole većem računskom centru prenošenje softvera i (naročito) podataka preko trake je masovna pojava što PC-ju otvara mnoga područja primene Cena ne žalcet baš i ne ohrabruje 1 300 dolara

Masovna memorija

Hard disk za „amigu 500"

Zašto bi samo vlasnici „amige 2000" uživeli u blagodetima hard diska? Obratite se firmi *Supra* (američki telefon ((503)967-9075) i za 995 — 3995 dolara nabavite hard disk od 20, 30, 60 ili 250 megabajta koji je dopunjen interfejsom za „amigu 500" Brzine prenosa podataka je preko 250 kilobajta u sekundi Kupovina hard diska je i prilika za memorijsko proširenje od 1 ili 2 megabajta koje se nalazi u istoj kutiji i napaja iz istog izvora



Šta ima novo

Komercijalni softver

dBASE IV ZA OS/2

Ashton Tate je najavio novu generaciju baza podataka koja će biti kompatibilna sa novim Microsoftovim operativnim sistemom OS/2. O karakteristikama se još ćuti, ali se može očekivati da će dBASE IV konačno obezbediti pozadinska pretraživanja, sortiranja i kreiranja izveštaja i ponuditi brojna unapređenja ionako sasvim solidnog programskog jezika.

Komercijalni softver

Konačno *SideKick Plus*

Dugo očekivani Borlandov *SideKick Plus* je konačno na tržištu! Za neočekivanih 200 dolara dobijate rezidentni tekst editor, kalkulator, kalendar, ASCII tablicu, komunikacioni paket i uslužni program za rad sa datotekama — skoro sve što vam usred rada na nekoj aplikaciji može zatrebati. Noviteta ima toliko da je nekadašnje uputstvo od 86 strana gotovo upetostručeno!

Da li smo nekada objavili Borlandovu adresu? Možda i nismo. *Borland International, 4565 Scotts Valley Dr, Scots Valley, CA 95066, USA.*

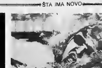
Proširenja

Noviteti mikro kanala

Vlasnici IBM-ovih računara iz serija PS/2 koja muče disketa od 3.5 inča zaista mogu da biraju — sve više nezavisnih proizvođača hardvera nudi dodatnu disk jedinicu od 5.25 inča! AT kompatibilne disk jedinice od 1.2 megabajta koštaju po 300 dolara i mogu da se nabave od firme *Pacific Rim Systems* (američki telefon ((415)782-1013) ili *Systems Manufacturing Technology* ((619)774-3590).

Nama zime ni za ljubitelja telekomunikacija: firma VenTel Inc (američki telefon ((800)538-5121) za 550 dolara prodaje modem VenTel 24/2 koji se ugrađuje u kutiju PS/2, obezbeđuje prenos podataka brzinom od 300 ili 1200 bauda i savršeno sarađuje sa programom *Crosstalk XVI*.

Ako vam svi ovi noviteti nisu dovoljni na Micro Channel konačno možete da priključujete i obične PC odnosno AT kartice — potrebna vam je samo tabla koju za 700 dolara prodaje *Advanced digital Corporation* ((714)891-4004). Nevolja je jedino što je za svaku PC karticu potreban po jedan *Run PC Bus* adapter.



Komercijalni softver

Još malo PC grafike

Zar ovo ne liči na fotografiju? Radi se ipak o kompjuterski dizajniranoj slici nastaloj primenom novog DeluxePaint programa za IBM PC firme Award winning software (britanski telefon (753)494-442). Program je Herkules, EGA, VGA, CGA i Tandy kompatibilan i obezbeđuje razne rotacije i, što je svojevrstan novitet, crtanje u perspektivi ili izometriji.

Štampač

Laserski kvalitet za manje od 1000 dolara

Laserski štampači su „ušli u modu" kada im je cena probila magičnu granicu od 2.000 dolara, iako cena i dalje padaju, dan kada će laserski štampač koštati manje od 1.000 dolara je još daleko. U međuvremenu možete da potražite vrlo kvalitetnu alternativu koju nudi renomirani *Hewlett-Packard* — HP DeskJet je štampač zasnovan na „pljuckajućoj" tehnologiji koju smo predstavili u prošlim „Računarima" i koji, nudeći „lasersku" rezoluciju 300·300 tačaka po inču i brzinu od 120 znakova u sekundi, košta svega 995 dolara. Iako su *bubble jet* štampači ubedljivo slabiji od laserskih kada se zahteva bojenje veće površina crnom bojom, kvalitet teksta i „pozitiv" slika je gotovo savršen.

Obrada teksta

Rečnik stranih reči

Pod stranim jezicima ovde podrazumevamo jezike koji nisu engleski — svi znamo da je jedan ne-engleski znak kao što je naše slovo č sasvim dovoljan da „sruši" brojne opcije tekst procesora od pretvaranja velikih slova u mala pa do deljenja reči na slogove. Pokazalo se, osim toga, da rezidentni rečnici o kojima se govorilo još od pojave personalnih računara predstavljaju prilično tvrd orah: kapaciteti memorija i diskova su i dalje premali da bi zamenili pristojan papirni rečnik. U tom svetlu *Term Tracer* firme *Ink International (P.O. Box 5477, 1007AL Amsterdam, The Netherlands)* treba smatrati interesantnim pokušajem: za 175 dolara dobijate anglesko — francusko — nemačko — holandsko — španski rečnik sa nekih 2500 reči koje su u svakom trenutku pristupačne. Tu je, naravno, i mogućnost kreiranja sopstvenih rečnika, što znači da ne treba gubiti nadu da ćemo jednoga dana na ekranu PC-ja videti i naše računarski sređene izraze.

Mipro '88.

Najmasovniji i najsvestraniji jugoslovenski skup iz područja mikroelektronike (MIPRO 88) infrastrukture celog korpusa visokih tehnologija održava se od 16. do 20. maja u kongresnom centru hotela „Adriatik" u Opatiji. Nakon desetogodišnjeg izuzetno uspešnog rada okupljajući svake godine 700—800 vrhunskih domaćih i stranih naučnika i stručnjaka MIPRO (mikroelektronika) izrastao je u jedinstveno žarište informatičke revolucije i postindustrijskog društva odakle se uticaji visokih tehnologija šire po celoj našoj zemlji.

Prema rečima inž. Mihajla Filiterovića, potpredsednika Organizacionog odbora MIPRO 88, savetovanja i seminari koji se tih dana održavaju u Opatiji imaju za cilj edukativno-stručnu podršku udruženom radu i obrazovnim institucijama jer su mentori i predavači najpoznatiji jugoslovenski naučnici. Savetovanja su: Mikroračunari u sistemima procesnog upravljanja (PU), Mikroprocesori u telekomunikacijama (TE) i Nove generacija računara (NG). Ovo poslednje uvek privlači posebnu pažnju jer predočava buduće horizonte mikroelektronička revolucije. Seminari su: Osnovi rada i programiranja mikroračunara (USM), Mikroelektronički sklopovi (UST) i Primena mikroračunara za nadzor i upravljanje tehničkih sistema i procesa (NSP). Uz ove organizuju se prikazi najboljih studentskih radova i izložba domaće i strane mikroelektronička opreme. Tema ovogodišnjeg okruglog stola jeste: Sadašnji privredni i ekonomski trenutak Jugoslavije i realni uslovi i putevi za razvoj visokih tehnologija.

Za bliže informacije treba se obratiti DSEIT Rijeka kao glavnom organizatoru na telefon 051/30-491 ili 30-082 (Trg Palmira Toljatija 4/I, 51000 Rijeka).

(S.M.S.)

Korak u 21. vek

Mladi Jugoslavije su se potrudili da ove godine umesto klasične ponesu štafetu znanja. U Novom Sadu se od 25. maja do 5. juna održava multimedijalna manifestacija „Korak u 21. vek", čiji je pokretač Narodna tehnika Vojvodine (mesto održavanja Novosadski sajam).

Prema rečima Đure Babića, sekretara NT Vojvodine, manifestacija je podeljena u nekoliko celina: Naučno-tehničko stvaralaštvo mladih (današnji mladi stvaraoci novog doba), Dani nauka i informatika, Sajam visokih tehnologija, Naučno-stručne tribine i paralelne aktivnosti. Sajam visokih tehnologija kao istinski korak u 21. vek predstaviće sve najznačajnije projekte za budućnost domaćih instituta, fakulteta i proizvođača kao i pojedinih stranih izlagača, a obuhvatiće mikroelektroniku, računarstvo, fleksibilne tehnologije, robotiku, nove materijale, biotehnologiju, kosmičku tehnologiju, nove energetske izvore itd.

Uz ovu i izložbu informatičke opreme sve vreme će se održavati pojedine takmičenja mladih u znanju i stvaralaštvu, prikazi njihovih radova, ekspozicija naučno-stručne literature i časopisa, naučno-tehnički bioskop, satelitska i kablovska televizija, mitinzi ekologista itd.

Iako je tek nedavno najavljena za manifestaciju „Korak u 21. vek" vlada ogromno interesovanje, tako da organizatori očekuju oko milion posetilaca iz cele zemlje. Nema sumnje mladi će biti veoma zadovoljni novom „štafetom znanja" koju će sami kreirati.

(S.M.S.)

Umetnost

Crtati kao kompjuter

Slike koje vidite nisu rezultat rada računara. Nije po sredi damp ekrana niti fini rad plotera. Iscrtavane su tušem rukom i rapidografom na hameru velikog formata. Odnos mašine i čoveka kao rezultat vremena u kome živimo ovekovečen je u likovnom radu Slobodana Kovačevića čiji crteži predstavljaju izvanredan primer „computer-like-art-a".

Slobodan Bobiš Kovačević, rođen u Beogradu 1957. godine, inženjer informatike, počeo je još kao gimnazijalac da se bavi izradom crteža u tušu koji ne pripadaju ni jednom od poznatih načina prezentacije, odnosno tehnika izrade. Samostalno je izlagao juna 1986. godine u Galeriji Osnovne organizacije Saveza sindikata Izvršnog veća Skupštine SR Srbije, uz mnogobrojne grupne prezentacije u okviru grupe „Treća dimenzija", nedavno preimenovane u Udruženje slobodnih umetnika „Đura Jakšić".



Preplitanjem bezbrojnih snopova uglavnom pravih linija, uz izuzetnu preciznost koja se graniči sa perfekcijom, Bobiš stvara veoma skladne dinamične kompozicije. Uslovno nazvani „ornamentikom", njegovi crteži mogu da se svrstaju u op-art, pa se kao asocijacija neposredno nameće Vazarelijeva „Vega". Vazarelijeva dinamika proizlazi iz deformacije, dok Bobiš svoju dinamiku plete čistom geometrijskom formom, prepuštajući figurama da pričaju svoju priču. Prepušteni sami sebi, oblici grade svoj svet koji nije uvek pod neposrednom kontrolom autora. Možda bi dodatna deformacija unela više duha i humanosti u njegove crteže i doprinela prestanku rasprava o tome da li je po sredi umetnost ili ne.

Shvatite ovo kao poziv

IZAZOV 21. VEKA

U okviru nove koncepcije obeležavanja Dana mladosti organizuje se između 30. maja i 5. juna na Novosadskom sajmu Multimedijalni program „Korak u 21. vek". Intencija ovog programa da smišljeno afirmišemo dostignuća i stvaralački rad mladih širom Jugoslavije.

Želja organizatora je da na jednom mestu okupi kreativni potencijal mladih koji se do sada javljao sporadično, a isto tako bivao i zabeležen u javnosti. Teškoće u kojima se, kao društvo, nalazimo ponekad nas jednostrano okraću stabilizacijl dok je senzibilitetu i sposobnostima mladih primereniji IZAZOV i stvaralačka VIZIJA mogućeg.

Konkretno, na programu će se naći sledeće oblasti i sadržaji:

— BERZA SOFTVERSKIH ILI HARDVERSKIH APLIKACIJA
čiji su autori pojedinci, ustanove ili organizacije
— MAIL BOX „KORAKA U 21. VEKU"
radiće u toku čitavog dana. Otvoren je za sve one koji u vreme manifestacije imaju na raspolaganju računar, modem, telefon i ideju
— PRVI JUGOSLOVENSKI „TURNIR" RAČUNARSKIH VIRUSA
(možda imate i „vakcinu" protiv njih)
— TURNIR „PROGRAMA — BORACA"
— I SVE DRUGO ŠTO PREDSTAVLJA PLOD VAŠEG UMOVANJA U OVOJ OBLASTI

Radnim organizacijama koje se bave visokom tehnologijom širom su otvorena vrata „SAJMA VISOKE TEHNOLOGIJE", koji treba da bude svojevrsna produktivna spona između vrhunskih dostignuća nauke i razvojnih mogućnosti i potreba privrede. „Korak u 21. vek", do kojeg nam je toliko stalo, možemo napraviti samo zajedno.

Koncepcija Multimedijalne manifestacije u sebe uključuje i brojna paralelna događanja (predavanja, tribine, okrugle stolove, demonstracije, prezentacije i dr.) čija je programska okosnica dovoljno fleksibilna da uvaži i ideje koje nam budete sugerisali.

Svi koji misle da se mogu svojim idejama, projektima, softverskim ili hardverskim rešenjima uključiti i predstaviti na manifestaciji ili sajmu, detaljnije informacije mogu dobiti pismom na adresu:

Savez mikroračunarske tehnike i
informatičara Vojvodine
21000 Novi Sad
Trg Lenjina 10/II ili na telefon (021) 25-499

Rukavica nam je bačena. Shvatite ovo i kao IZAZOV

Redakcija „Računara" i Institut „Mihajlo Pupin"

Samogradnja „tima 011"

Mali predah

• Akcija „Napravi i ti tim 011" je u ovom trenutku definitivno potvrđena. Iako čitaocima dok pišemo ove redove nisu poznate ni dokumentacija ni uslovi kupovine, stiglo je mnogo preliminarnih narudžbenica. Poštari nam donose čak i kupone za nagradnu igru lista „Tim"! U trenutku zaključivanja ovog broja naše specijalno izdanje je još uvek u štampi i pravo interesovanje očekujemo tek pošto se čitaoci detaljnije upoznaju sa „timom". Još jednom napominjemo da cena zavisi od vašeg interesovanja i to — obrnuto proporcionalno.

• Mnogi čitaoci se interesuju za postupno naručivanje delova za samogradnju. Moći ćete da naručujete delove onim tempom koji vam odgovara. Osnovni komplet sadržavaće štampanu ploču i sistemsku disketu. Konačna cena štampane ploče biće utvrđena 15 dana po izlasku specijalnog izdanja na osnovu broja prispelih narudžbenica. Cena sistemske diskete biće niža nego cena prazne diskete u maloprodaji. Diskete nećemo prodavati odvojeno, jer se bojimo interesovanja vlasnika ostalih računara. Programiranje EPROM-a naplaćivaće se simbolično.

• Mala i velika privreda pokazuju veliko interesovanje za našu akciju, što će, verujemo, doneti nova pojeftinjenja gradnje.

• Na sistemskoj disketi se za sada nalazi operativni sistem ZDOS, GBasic v5.22, editor teksta „Wordstar 3.0", Borlandov „Turbo Pascal", monitor, dva asemblera (jedan od njih je makroasembler), gomila korisnih programa za instaliranje i popravljanje sistema (print, kopiranje diska, čeprkanje po disku i memoriji...).

• Obećavamo da nećemo biti mirni dok ne napunimo sistemsku disketu. Ako treba, pisaćemo sami.

Izložbe

Kompjuter '88

Od 7. do 11. aprila u beogradskom Domu omladine održana je priredba pod nazivom „Kompjuter '88". Na priredbi je učestvovalo šesnaest izlagača iz zemlje i inostranstva. Žao nam je što taj broj nije bio veći. Razlog je, verovatno, dosta kasna najava sajma, pa proizvođači nisu stigli da unesu izložbu u svoj plan. Posebno interesovanje vladalo je za ponudu „Mraz Elektronika" iz Minhena koji je nudio kompatibilce po dosta povoljnim cenama. Nadamo se da će na narednim priredbama veći broj proizvođača uzeti učešća, tim pre što smo do prisutnih izlagača čuli da je izlaganje bilo korisno. Priredbi želimo da postane tradicionalna, jer je beogradskoj publici potrebna ovakva specijalizovana izložba, a organizatorima, Domu omladine Beograda i časopisu „Svet kompjutera", preporučujemo manju selektivnost prilikom slanja prijavnica.

Pet godina uvođenja računara u škole

Natezanje živaca

Kada smo, sredinom 1982, pripremali „Računare u vašoj kući", smatrali smo da će kompjuteri koliko sutra ući u školske klupe. Bila je, čak, aktuelna i šala da je to odličan način da klinci omrznu računare: naterajte ih da o njima uče u školi, dajte im četiri pismena zadatka godišnje i sjajna zabava će se pretvoriti u bauka! Došla je i 1988, a deca i dalje vole računare — kako je dinamika njihovog uvođenja u škole krenula, nije nemoguće da će ih voleti i 2000. godine.

Prvi i verovatno najozbiljniji problem koji otežava uvođenje računara u škole je svakako izbor školskog kompjutera. Poneko će možda reći da su izbor školskog računara iskomplikovali proizvođači domaćih i domaćih kompjutera ali se mi sa takvim tvrdnjama teško možemo složiti — ne treba kriviti proizvođače računara što se bore za svoje interese, ali treba uputiti mnoge zamerke školskim vlastima koje se postavljena između zahteva raznih proizvođača ponašaju kao onaj čuveni magarac između dva plasta sena!

Od samog početka je bilo jasno da bi najracionalnije, najjeftinije i uopšte najbolje rešenje bilo izbor **jugoslovenskog** školskog računara: domaći hardverski i softverski potencijali nisu naročiti i ludo je zahtevati da se itekako složeni problem pisanja školskog softvera nezavisno rešava na osam ili čak na dva različita mesta. I pored toga svakom vlasniku pomanute elementarne doze inteligencije bilo je jasno da od jugoslovenskog školskog računara nema ništa — zar bi se moglo zamisliti da jedna suverena država kao što je Srbija uvozi školske računare iz Hrvatske ili Slovenije (važi, naravno, i obrnuta logika) i tako deo svog teško izdvojenog prosvetarskog dinara poklanja inostranim proizvođačima kompjuterske opreme? „Tuđi" školski računar bi osim toga značio da u „tuđoj" sredini živa bolji stručnjaci za računare, što je nazamislivo — svaka će republika i pokrajina pa i svaka opština dakle imati svoj školski kompjuter!

Ko dira — pokvariće

Čak ni u okvirima jedne republike stvar ne teče glatko — sredinom 1983. godine prosvetne vlasti Srbije intenzivno su se trudile da izaberu školski računar. Konkurencija je bila stvarno velika: sa jedne strane „galaksija", čiju je proizvodnju organizovao Zavod za udžbenike i nastavna sredstva, a sa druge „lola 8" koju je „gurale" beogradska fabrika „Ivo Lola Ribar". Nije nam poznato koliko je sastanaka održano i koliko je kafe, kiselih vode i ostalih pića popijeno, poznato je, međutim, da prosvetne vlasti nisu mogle da zaključe koji je računar bolji, pa su odlučile da svakoj školi kupe po jednu „galaksiju" i jednu „lolu 8", naravno bez ikakve periferija opreme. Obzirom da je teško zamisliti dva međusobno nekompatibilnija računara, nikakav se školski softver nije mogao razvijati.

Ukoliko u školi nije bilo nekog ambicioznog nastavnika koji bi pokušao da pokrene dobijene računare, oni su ostajali u vitrinama, što samo po sebi školama uopšte ne smeta — ako ste nekada bili dragi gost obližnje škole, znate da „su u ovoj vitrini priznanja koje je naša škola dobila na takmičenju u poznavanju saobraćajnih propisa, u „ovoj vitrini plaketa koju su 1953. godine dobili naši gorani, u ovoj vitrini naš herbarijum i zbirka minerala, u ovoj vitrini maketa nove školske zgrade koja će svakako biti izgrađena pre 2100. godine", a u „onoj vitrini kompjuteri koje naša škola poseduje". Samo se po sebi razume da se višestruki katanci koji sprečavaju pristup svim ovim dragocenostima redovno skidaju radi brisanja prašine.

Bilo je, na sreću, ambicioznih nastavnika matematike ili fizike koji su u želji da pokrenu školske „galaksije" i „lole" od kuće donosili monitore, kasetofone, kablove... Takvi bi se brzo razočarali — potpuno ispravan računar verovatno nije dobila ni jedna škola, ako je sve drugo baš i bilo ispravno, veza računara i kasetofona svakako bi se pokazala veoma nepouzdanom! Proizvođači računara su uvek imeli opravdanje — školski računari su morali da budu toliko jeftini da je pravo čudo što je i takve verzije obezbeđena uz zakašnjanje od samo tri meseca, ako veza sa kasetofonom ne funkcioniše, kriv je naravno kasetofon! I tako su nastavnici bili upućeni na razne servise koji su pomagali samograditeljima „galaksije" — čak i ako bi svoj računar nekako osposobili za rad, pokazalo bi se da je jedan kompjuter sasvim nedovoljan za osnivanje ma kakve kompjuterske sekcije — šta da radi pedeset ili sto učenika sa jednim računarom? Tu je najzad i krunski argument koji redovno potežu nadležni direktori, zamenici direktora, domari i slični — nije TO za toliku decu, oni će TO začas pokvariti! I tako „galaksije" i „lole" postaše računari kompatibilni sa (u našim školama veoma zastupljenim) operativnim sistemom „vitrijaner".

A sada ozbiljno

Pošto se prvi pokušaj uvođenja računara u škole ovako slavno završio, prosvetne vlasti su ponovo okupile i zaključile da „ovoga puta stvarima treba ozbiljno prići". Ozbiljan pristup podrazumeva formiranje kriterijuma za izbor školskog računara, raspisivanje konkursa, detaljno testiranje predloženih kandidata i najzad izbor školskog kompjutera. Što se definisanja kriterijuma tiče, posao je uspešno završen: zaključeno je da „kompletan hardver i softver treba da budu plod domaćeg razvoja" i „da računar treba da bude kompatibilan sa nekim od afirmisanih svetskih standarda" (nije baš jasno kako se kompatibilnost sa svetskim standardima usklađuje sa domaćim softverom, ali... niko nije savršen). Kandidati su se brzo javili i trebalo je još samo izabrati školski računar. Ni tu nije bilo problema: školski računar je izabran, i to nekoliko puta!

Uz rizik da propustimo neku fazu, reći ćemo da je školski računar „galaksija plus", i da je školski računar „oric" — ambiciozni proizvođač je uspeo da ubedi prosvetne vlasti da je računar u stvari domaći, pošto je po njegovom fijasku na svim stranim tržištima otkupljen kompletan inostrani projekat. Sigurna ali odvratna stvar — u škole ide „pekom". Ne, u škole idu PC kompatibilci. Ne više (nisu plod domaćeg razvoja) — u škole ipak ide „pekom". A tu je i TRS Color Computer, pošto je 2000 primeraka ovog računara budzašto prodato nekoj crnogorskoj firmi. Ma ne, u škole ide „tim 011". Pa ne baš u sve — u neke ipak ide PC. A u nekima je već „pekom". A „oric" je jedini računar koji se može nabaviti u prodavnicama... Razne „izabrane" školske računare mogli bismo u stilu Nela Raulison nabrajati danima — objektivno posmatrano, Srbija trenutno ima pet-šest školskih računara, što znači da u stvari nema ni jedan — stanje u većini ostalih republika nije mnogo bolje.

Gordijev čvor

Da li je univerzalni školski računar mogao da se proizvodi uz puno uvažavanje naših deoba? Mogao je, i to na više načina. Pomenućemo na primer ideju konstruktora „galaksije" Voje Antonića, koji je još 1983. godine smatrao da bi trebalo raspisati konkurs za jugoslovenski školski računar, izabrati jedan od primljenih projekata, otkupiti ga od autora i staviti na raspolaganje javnosti. Tako bi se školski računar proizvodio na više mesta, pa bi svaka škola mogla da izbegne odlivanje sredstava u drugu opštinu — konkurencija proizvođača bi osim toga blagotvorno uticala na cenu, a klinci zainteresovani za hardver mogli bi sami da sklope svoj računar.

Ako ovo zvuči previše komplikovano, mogao bi se slediti primer sa kojim smo se upoznali na nedavno održanoj tribini o uvođenju računara u škole koju je organizovao „Radio Beograd" u saradnji sa redakcijama „Računara" i „NIN"-a: jedan od prosvetnih savetnika kragujevačkog regiona rekao je da se prosvetni savet sakupio, proučio ponude proizvođača i odlučio da školski računar bude PC klon. Računari su potom naručeni, plaćeni i dobijeni. Reklo bi se da u čitavoj operaciji nema ničeg posebnog — prirodan način da se izabere školski računar. Tragično je, međutim, što je nas postupak kolega iz Kragujevca u pravom smislu fascinirao — zar Jugoslavija ili Srbija nije mogla tako da „sedne", izabere i nabavi školski računar? Ne kažemo da to mora da bude PC — neka školski računar bude „tim 011" ili „oric" ili „galaksija plus" ili čak „pekom", ali neka se jednom Gordijev čvor preseče!

324



— da bismo računare uveli u škole **moramo** da ozvaničimo jedan školski računar a to ozvaničenja kako izgleda nije ni na pomolu Prosvetne vlasti se kako rekosmo ponašaju kao ona životinja sa dugim ušima između nekoliko plastova sena

Čak i sasvim umeren cinik svakako će reći da je ovakvo ponašanje prosvetnih vlasti sasvim očekivano i da se sjajno podvodi pod sveobuhvatnu frazu naša posla Ono što u celoj stvari čudi su međutim proizvođači računara — nikako nam nije jasno zašto im je toliko stalo da se probiju u škole! Proizvodnja školskog računara je prema onome što se gledanjem sa strane može zaključiti očajan posao tržište jeste veliko, ali su škole i prosvetni savati puka sirotinja što znači da cena školskog računa jedva-jedvice pokriva troškove proizvodnje uvećane za nezaobilazni šticung deviza Školski računar je jasno neka vrsta reklame i investicija u budućnost ali se druge strane ko može tvrditi da se i neka velika fabrika ili institut za nekoliko godina neće boriti za goli opstanak izvozeći korpe od pruća?

Računari i OTO

Paralelno se izborom školskog računara već pet godina traje uobličavanje nastavnog plana i programa Najprirodnije bi jasno bilo uvesti poseban predmet koji bi se zvao Računarstvo ili Računarstvo i Informatika ali se pokazuje da taj predmet nije baš lako udenuti u sedmicu koja uprkos svim željama i odlukama ima svega 168 časova Ako negde izreknete tvrdnju da se u školi uči i radi mnogo sasvim nepotrebnih stvari sve je prilika da će se svi sagovornici složiti Ukoliko se međutim neko usudi da predloži smanjenje fonda časova maternjeg jezika fizičkog vaspitanja geografije domaćinstva ili nekog drugog predmeta začas će se pojaviti bezbroj akata (na svakom od njih po neki veliki pečat) u kojima piše da i ovoliki fond časova nije dovoljan da spreči da deca budu nepismene rahitična, neupoznata sa osnovnim planinskim masivima i nesposobna da skuvaju supu iz kesice Tu je naravno i krunski argument da bi se smanjivanjem fonda časova nekog predmeta pojavio veliki tehnološki višak nastavnika — taj argument potvrđuje večnu istinu da učenici i, uopšte škole postoje zato da bi nastavnici primali plate!

Pošto se fond časova pojedinih predmeta ne sme smanjivati i pošto se produženje redne nedelje jednostavno ne bi moglo podnati ostalo je da se računarstvo uda za neki postojeći predmet Izbor je pao na OTO ili punim imenom Opštetehničko obrazovanje OTO je sjajan predmet u okviru koga je autor ovoga teksta pre desetak godina naučio sledeće velike životne istine (napomena dežurnom lektoru nama je ovako diktirano)

1 Prilikom fotografisanja na filmu se dobija negativna fotografija koja se docnije izazivanjem pretvara u pozitivnu — pozitiv

2 Za umetanje burglije u rotirajuću bušilicu treba koristiti zvezdasti ključ koji se uz bušilicu isporučuje Isti ključ može da umetne razne burglije primenom tehnika različitog zatezanja

3 Sifon ne služi da bi se u njemu zadržavala prljava materija Sifon služi da se spreče neprijatni mirisi da se ne vraćaju u stan

Verujemo da će u okviru OTO-a računari biti predstavljani na sličan način mnogo velikih istina koje će bez sumnje biti ispisane tehničkim slovima u svesci bez linija čiji je svaki list obeležen marginom (12 mm sa leve i 8 mm sa desne strane) Zgodna strana OTO-a je što obezbeđuje izvanrednu proveru savladanog znanja — najvišu ocenu uvek dobija najbolji poznavalac računara koga nije teško prepoznati po najurednijoj svesci!

U nebranom grožđu

Pošto većina nastavnika OTO-a u okviru školovanja nije imala prilike da upozna računare prošlog leta su organizovani 20-časovni kursevi koji su pouzdano popunili ovaj propust — novoproizvedeni računarski stručnjaci će znanja koja su na ovim kursevima stekli i utvrdili bez sumnje uspešno preneti učenicima, pa će za nekih desetak godina neki budući saradnik ,Računara u svojoj svesci za OTO svakako pronaći sledeće velike istine

4 Računar se obično hvata na 36 kanalu ali se na televizorima Ei Niš koji imaju samo osam kanala mora hvatati na osmom koji se prethodno podešava

5 Računanje u dvostrukoj tačnosti je sporije od računanja u standardnoj tačnosti pošto se svako računanje vrši dva puta radi provere

Ne treba se naravno ljutiti na nastavnika OTO-a koji su u ovoj situaciji najmanje krivi Oni su se i sami našli u nebranom grožđu da predaju jedan predmet pod nemogućim uslovima za koji su na silu instant tehnikom prekvalifikovani

Dok bi se izbor školskog računskog predmeta dao brzo i racionalno rešiti nastavna računarstva ni iz daleka nije takva — nikakva čarobna formula ne može da obezbedi malim školama u malim mestima kvalifikovane predavače! Verujemo ipak da bi u nedostatku boljeg rešenja i vanbračna veza računarstva i matematike dala bolje rezultate nego brak informatike i OTO-a!

Što se nastavnog plana i programa tiče stvar je još luđa Na svakoj tribini u svakom nastupu i okviru svakog referata svi se slažu da se treba baviti **korišćenjem** a ne **programiranjem** Razlozi su sasvim jednostavni već je u ovom trenutku broj korisnika računara višestruko veći od broja programera a u budućnosti će se taj raskorak i dalje povećavati — romantični računari koji postoje da bismo ih programirali sve više prelaze u vredne alatke neophodne za svakodnevni život I pored tako nepodeljenog mišljenja u osnovnim školama će se učiti bejzik ili paskal a ne dBASE III Lotus — niko ne zna *zašto* će tako biti ali svi znamo da će biti baš tako

Ko radi

Iako je već pet godina aktuelno uvođenje računarstva u škole nije daleko odmaklo — nemamo školski računar nemamo školski softver niti softversko tržište nemamo definisan predmet nemamo prihvatljiv plan i program nemamo nastavnika i profesora Jedina uteha je da su konačno svi shvatili da će računarska pismenost uskoro biti potrebna koliko i Vukova azbuka i da se pred uvođenje računarstva u škole ne postavljaju nikakve formalne prepreke

Ako su narodne poslovice ko radi taj i greši i na greškama se uči tačne naše prosvetne vlasti su se u toku ovih pet godina naradile i naučile mnogo toga. Vreme je da se tako slavno stečeno znanje ponegde i primeni!

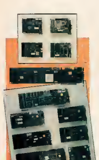
Dejan Ristanović

22 EGA kartice

Lepe kao san

Iako se često tvrdi da je IBM, dizajnirajući svoj personalni kompjuter, napravio samo jednu kapitalnu grešku zvanu grafika, činjenica da je PC računar sa najviše grafičkih standarda u istoriji počinje da pokazuje i svoja dobra lica. Ukoliko, naime, u jednom trenutku prestanete da budete zadovoljni grafikom vašeg „spektruma", „komodora", „amstrada" ili BBC-ja, ostaje vam samo da kupite novi računar. Ako, sa druge strane, grafika PC-ja više ne zadovoljava vaše potrebe, promenićete samo video karticu i, eventualno, monitor. Dok za jugoslovenske PC-jevce čak i EGA kartica predstavlja poprilično luksuz, svetski trendovi su poodavno prevazišli ovaj standard — hit poslednjih meseci su proširena EGA i VGA kompatibilna kartice, koje ćemo predstaviti uz pomoć napisa Enhanced EGA and VGA Boards koji je objavljen u martovskom „Bajtu".

Premda u početku oštro kritikovan EGA standard je osvojio brojne korisnike — iako su se mnogi „kleli" da monohromatski monitor koriste zato što je zdraviji za oči, poboljšanje specifikacije kolor-monitora praćeno padom njihovih cena učinilo je da brojni PC-jevci shvate da monitor u boji ipak nudi mnogo više: jednostavnije isticanje teksta, očigledniju poslovnu grafiku, lakše projektovanje višeslojnih štampanih pločica i tako dalje. Kada jednom prežalite cenu kolor monitora, nema nikakvog smisla prihvatiti siromašni ispis teksta koji nudi CGA kartica, što znači da EGA predstavlja minimalno rešenje. Njena grafika 640•350 u 16 boja će verovatno zadovoljiti sve vaše potrebe: horizontalnih 640 tačaka je sasvim solidna cifra, a ni na vertikalnih 350 tačaka ne smemo mnogo da se žalimo — pre svega godinu-dve jeftin računar koji nudi grafiku 640•256 predstavljao je svetsko čudo!



Od svega pomalo

Tristo pedeset tačaka u koloni na žalost ipak ne ukida „testeraste" kose linije — ima se utisak da bi se samo malim povećanjem vertikalne rezolucije mnogo dobilo na kvalitetu slike! Toga su se dosetili brojni proizvođači računara i tako su nastale poboljšane *(enhanced)* EGA kartice koje nude grafiku 640•480 u 16 boja. Moglo bi se reći da postoje samo dve suštinske razlike između EGA i poboljšane EGA kartice: jedna je frekvencija oscilatora (većina klon-kartica radi na frekvenciji koja prevazilazi EGA-inih 16 257 MHz) a druga, prosto rečeno, savršena slika. Konkurentski grafički adapteri su se po ceni sasvim približili standardnoj EGA kartici što znači da je eventualni nedostatak kompatibilnosti jedini razlog koji može da vas usmeri na original.

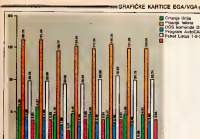
IBM je u međuvremenu predstavio svoje računare iz serije PS/2 i, zajedno sa njima, promovisao VGA grafički standard. VGA uvodi tri nova grafička moda čije su oznake 11H (640•480 u dve boje) 12H (640•480 u 16 boja) i 13H (320•200 u 256 boja), kao i dva nova tekst moda koja su naročito značajna za poslovne primene. Bitna je i promena monitora — IBM-ovi inženjeri su izračunali da bi kablovi koji povezuju PS/2 sa TTL monitorom mogli da budu dugi svega nekoliko centimetara, pa su prešli na tehniku analognih ekrana, što većini korisnika nalaže kupovinu nekog od multisinhronih monitora o kojima smo govorili u „Računarima 35".

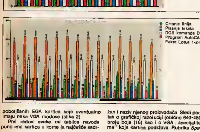
Raznim konstruktorima EGA kartica veoma su se dopali novi VGA modovi, pa su se na tržištu pojavili i prvi klonovi. Treba, međutim, razlikovati poboljšane EGA kartice koje „slučajno" imaju neki od VGA modova od pravih klonova: minimalni uslov koji čini da kartica bude VGA kompatibilna je potpuna emulacija poziva BIOS-a, a veoma je poželjno da se uspostavi i hardverska kompatibilnost na nivou registara. Zato smo kartice o kojima govorimo podelili u dve grupe — osam pravih VGA klonova koji su opremljeni (skoro) svim modovima i imaju kompatibilni BIOS (slika 1) i 14

VGA KARTICE

Model	ATI VIP VGA	Tatung Platinum Card	IBM PS/2 Adapter	Sigma VGA	STB VGA Extra
Maksimalna rezolucija	640•480	640•480	640•480	640•480	640•480
Boja	16	16	16	16	16
VGA modovi	Svi	Svi	Svi	Svi	Svi
Specijalne funkcije			Podržava Herkules EGA i CGA mod.		Video ROM drži u RAM-u
Dokumentacija	49	29	43	63	57
Cena (dolara)	450	450	600	500	395
Model	Video Seven VEGA VGA	Zenith Z449	Compaq Video Graphic		
Maksimalna rezolucija	640•480	640•480	640•480		
Boja	16	16	16		
VGA modovi	Svi	Svi osim 13H	Svi		
Specijalne funkcije		Podržava Herkules EGA i CGA mod			
Dokumentacija	52	24	29		
Cena (dolara)	500	500	600		

slika 1





poboljšanih EGA kartica koje eventualno imaju neka VGA modove (slika 2)

Prvi redovi svake od tabilca navode puno ima kartice u kome ja najčešće sadržan i naziv njenog proizvođača Sledi podatak o grafičkoj rezoluciji (obično 640•480) i broju boja (16) kao i o VGA „specijalitetima" koja kartica podržava. Rubrika *Specijalne funkcije* odnosi se na dodatke koji su uključeni u cenu — negde se dobija miš negde serijski ili paralelni interfejs i tome slično U *specijalne funkcije* uključili smo i sposobnost nekih kartica da emuliraju „stare" standarda kao što su EGA, CGA ili Herkules Preostela dva reda sumiraju cenu kartice i obim dokumentacije koja se uz nju dobija

Proširene EGA kartice						
Model	**Boca Research Mul EGA**	**Genoa Super EGA HIRES+**	**IGC EGA card**	**Logitech EGA**	**Mitac SEGA**	**NSI Logic Smart EGA+**
Maksimalna rezolucija	640•480	640•480	640•480	640•480	640•480	640•480
Boja	16	16	16	16	16	16
VGA modovi	—	0,1,2,3	—	—	—	—
Specijalne funkcije			TTL i analog izlaz.	Cena obuhvata miša.		Kompatibilna sa drajverima za običnu EGA karticu.
Dokumentacija	46 str	57 str	25 str	47 str	26 str	39 str
Cena (dolara)	300	490	400	400	200	500

Paradize Autosw EGA 480	Quadram Duad ProSync	SCOA Star PGA	SMT ProEGA	Tatung 900 OmniCard	Tecmar EGA 900 Master	Thomspon EGA Ultra Version	Tseng Labs EVA480
640•480	640•480	640•480	640•480	800•560	640•480	640•480	640•480
16	16	16	16	16	16	16	16
—	11H, 12H	—	—	11H, 12H	11H, 12H	—	—
						Microsoft InPort konektor	RS—232C Herkules emulator
86 str	52 str	60 str	60 str	70 str	70 str	44 str	51 str
350	400	400	250	400	500	400	480

slika 2

Poboljšana VGA

Pošto su VGA kartice glavna tema ovoga teksta pokušaćemo da ukratko predstavimo modele pobrojane u tabeli 1 — parametri svih modela su relativno slični što znači da se na osnovu same tabela teško može doneti ispravna odluka

ATI VIP VGA i **Tatung Platinum Card** su praktično identična kartice koja po istoj ceni prodaju dve različite kompanije čak je i propratna oprema ista — uz svaku se karticu dobija identično uputstvo kao i drajveri za *GEM Venturu Lotus AutoCAD* i *Windows* Kvalitet drajvera nije baš slavan slika je previše „rastegnuta" pa se prvi i poslednji redovi često ne vide što praktično onemogućava primenu *AutoCAD-a* Moguće je naravno da će se u budućnosti pojaviti bolje verzije drajvera ali je do tada kupovina bilo koje od ovih kartica nepreporučljive

Nesumnjiva prednost **IBM-ove PS/2 VGA** kartice je dakako stoprocentna VGA kompatibilnost Ako se izuzma nedostatak drajvera koji je u našoj piratskoj sredini lako rešiv kartica predstavlja izvanredan izbor ali joj cena svakako nije ohrabrujuće

Sigma VGA je, poput IBM-ove VGA kartice CGA EGA i Herkules kompatibilan adapter, u cenu od 500 dolara uračunati su i drajveri za *AutoCAD Lotus* i *MS Windows* Kartica je u negativnom smislu slična **Tatung Platinum** kartici — poslednji red teksta često ne staje na ekran!

STB VGA Extra je najjeftinija ali na žalost, i najslabija kartica koju pominjemo Ne samo da je ispis gotovo redovno veći od ekrana već se i „viškovi" preklapaju čineći deo teksta nečitljivim Priloženi drajveri za razne programa se čudno ponašaju pa se po napuštanju neke aplikacija računar često mora resetovati

VEGA VGA kartica je kraća od ostalih što znači da zauzima samo „poluslot" Uz karticu se isporučuje softver koji prepisuje kompletan sadržaj sistemskog ROM-a u RAM što začuđujuće ubrzava rad U cenu su uračunati i drajveri za *AutoCAD MS Windows* i *Lotus* Cena je, kada je već pomenusmo, unekoliko visoka ali je kartica inače vredna preporuke

Zenith Z449, za razliku od svih ostalih kartica, ne podržave jedan od VGA modova — nedostaje, da bi stvar bila posebno neprijatna, mod 13H koji je sa svojih 256 boja pravi specijalitet VGA standarda! Za uzvrat tu je pogodna Herkules, EGA i CGA kompatibilnost, kao i drajveri za *AutoCAD, GEM Lotus Symphony* i *MS Windows* i to na disketama od 5 25 i 3 inča!

Compaq Video Graphics Controller Board je pravi pravcati klon IBM-ove VGA kartice — konstruktori su imitirali čak i IBM-ov manir da uz karticu ne daje ni jedan jedini drajver! Karakteristika kartice su više nego solidne, ali se pojavljuje neprijatna interferencija sa zvučnikom — uz malo prakse na cenovu tihog i neprekidnog tona ▶

možete da pogodite šta se događa na akranu! Idaaino za korianike koji aiaod vide iii aiaod čuju

Konačni izbor se, dakia, svodi na originalnu IBM-ovu **VGA karticu, VEGA VGA i Compaq Video Graphics Controliar.** Ukoiiko to novčana sredstva dopuštaju, svakako ćemo se opredeiiti za IBM-ovu karticu kojoj prateći softver u budućnosti svakako neće nedostajati!

Poboljšana EGA

Pošto su sva kartice koja pominjamo u tabali ralativno slična, ograničićemo se da detaljnija opišemo najskupiju, najjeftiniju i karticu koja nudi najvišu razoluciju

Mitac SEGA kartica košta svega 200 doiara aii to ne znači da ima neku posebnu manu — radi sa o kvalitatnoj kartici koja nudi stabilnu siiku i raiativno brzu grafiku. Nevolja sa pojavljuju jedino kod nakih komercijalnih programa koji na mogu da se usaglase sa karticom pa dala akran na dve idantična i veoma usitnjene polovina. Pojavom odgovarajućih drajvera ovaj će se probiam svakako rašiti

Tecmar EGA Master 800 košta čitavih 600 dolara i to varovatno zbog smanjanih dimenzija (potreban ja samo jedan poiuslot) i delimično VGA kompatibilnosti — BIOS obezbeđuje VGA modove 11H i 12H. U cenu su uključeni drajveri za *MS Windows, GEM, Lotus, AutoCAD* i *Venturu*, kao i veoma kvalitetno uputstvo za instaliranja i upotrabu

Tutung 800 Omni Card nudi fantastičnu razoluciju 800•560 i to u programima *MS Windows, GEM, AutoCAD Lotus* i *Symphony* iako cena nije posebno visoka (400 dolara), kupovina ove kartice predstavija svojevrstan rizik — format akrana ja nestandardan, što znači da iako može da ostane bez podrška

Praporuke sve u svemu zaslužuju **Mitac SEGA** i **Paradize Autoswitch EGA 480** — za prvu ćemo se odlučiti ako su sredstva krajnja ograničena, a za drugu ako nam je potraban dobar softver, popularno uputstvo i solidan kvalitet izrada koji ipak nija uzdigao cenu do nebesa.

Na brzinskom testu

Iako ja hardver relativno aličan, neke su kartice pri radu značajno brže od drugih. Na slici 3 vidimo brzinske rezultata poodijšanih EGA kartica koja su se takmičila u crtanju linija, pisanju teksta, izvršavanju DOS komande DIR i crtanju posredstvom *AutoCAD-a* i *Lotus-a*. Brzinski šampioni su očito **Boca Research MultiEGA** i **Tseng EVA480.**

Na slične su iskušenja stavljene i VGA kompatibilne kartice (slika 4) — „pobedio" ja **Compaq-ov VGC** adapter dok je drugo mesto zauzela kartica firme **Video Savan.** IBM-ova kartica je, kao i obično, nagde u zlatnoj sredini!

Stalni napredak video kartica, bez sumnja, otvara nova polja primene personalnih računara, ali na treba zanemariti ni afakat koma se raduju korisnici plićeg džepa — svaka nova generacija navodi proizvođače „starih" CGA i Harkules klonova da obaraju cene što varovatno znači da će Harkules standard još dugo vladati našim meridijanima!

Priredio: Dejan Ristanović

Adem Jakupović

dBASE III Plus

Tehnička knjiga, Beograd, 1988
210 strana, 18 000 dinara

dBASE polako postaje prava medijska tama — gotovo da nama kompjuterski orijantisanog časopisa ili izdavača koji je propustio da svojim čitaocima ponudi manje ili više opširan dBASE priručnik! Zapaženo mesto u ovoj seriji svakako zauzima i dBASE III Plus *Adema Jakupovića* u izdanju beogradske Tehničke knjige

dBASE III Plus počinja sasvim klasično: uvodi se pojam baze podataka, opisuja njena organizacija i upotraba, ukratko nabrajaju dobra i loše strane dBASE-a i što je karakteristično za ovu knjigu, ukratko opisuju prateći programi *Clipper QuickCode III QuickReport DBC DBL i Applications Generator*. Opis složenijih programa je sasvim kratak, dok su jednostavniji (i za početnika značajniji) dodaci relativno detaljno opisani

Treći deo se bavi opisom problema koji će se kroz čitav ostatak knjige rašavati. Možda ja izabrani problem (fakturisanje) na prvi pogled prilično ododjan (asocira na nakakvo komplikovano knjigovodstvo), ali se u stvari radi o jednostavnoj stvari: traba ustanoviti dugovanja i ispisivati računa. Kroz ostala poglavlja će se pokazati da ja problem veoma instruktivan i da obezbeđuje demonstraciju mnogih mogućnosti dBASE-a, baza podataka je na kraju krajeva u osnovi i usmarana ka ovakvim primenama

Sledeća poglavlja bave se instalacijom dBASE-a, kreiranjem šema baze i unošenjem podataka. Zanimljivo ja da sa autor odlučio da praktično ignoriše program Assistant koji obezbeđuje korišćenje dBASE-a preko menija i koji je prema tome veoma pogodan za početnika i ljuda koji se profesionalno ne bave računarstvom. Od čitaoca se umesto toga zahteva da deaktivira ASSIST i onda kuca komanda dBASE-ovog jezika. Dobre strane ovakvog pristupa su što se čitalac stimuliše da što pre uđe u tajna dBASE-ovog programskog jezika i što kucanje komandi obezbeđuje delimičnu kompatibilnost sa ranijim verzijama dBASE III Plus kao što je dBASE II. *Ashton Tate* ja sa druge strane kreirao Assistant upravo da bi pomogao korisnicima koji se užasavaju pamćenja i kucanja komandi — verujemo da bi knjiga bila pristupačnija širem krugu čitalaca da je Assistant detaljno obrađen.

U okviru petog poglavlja opisuje se unošenje, ispravljanje, brisanje, uređivanje, pretraživanje i filtriranja slogova, priprema



maski, nalepnica i izveštaja i kao poseban specijalitet, rad sa više datoteka. Šesto poglavlje namenjeno ja programerima: opisani su programi za upisivanje, ažuriranje, fakturisanje i štampanja spiskova, a onda date ideje za proširanje aplikacija. Poslednja poglavlja ja uglavnom informativno i bavi se kompatibilnošću dBASE-a sa drugim softverskim paketima i radom u mreži

Značajan deo knjige (skoro 90 strana) zauzimaju dodaci u kojima su tabelarno prikazane i detaljno opisane komanda i funkcija dBASE-a, HELP biblioteka i konfigurisanje programa. Posebno su kompletna prva dva dodatka koja praktično sumiraju čitav programski aspekt dBASE-a III Plus

Knjiga dBASE III Plus ja profesionalno prelomljena, solidno tehnički pripremljena, prošarana umerenim brojem štamparskih grešaka i dopunjana (po našoj oceni) izuzetno inventivnom naslovnom stranom u kojoj se prepoznaje stil *Zorana Brankovića*

dBASE III Plus ja sve u svemu više nego dobrodošla dopuna vaše kompjuterske biblioteka — knjiga ja možda malo više namanjana programerima a malo manje korisnicima, ali verujemo da će čak i apsolutni početnik u njoj naći dosta korisnih informacija i tebela koje će često referencirati u docnijam radu. Šteta ja jedino što je porast cena unekoliko udaljio kompletnu kompjutersku literaturu (pa samim tim i ovu knjigu) od mnogih potencijalnih čitalaca i korisnika dBASE-a!

Dejan Ristanović

Komodor MPS 1200

Temperamentni mališa

Periferijske oprema „komodora" zbog svoje sporosti, visokih cena i teškoća sa povezivanjem nikada nije uživala naročiti ugled među ljubiteljima računara. Iako „komodor" ni ovoga puta nije odstupio od svoje filosofije, pojava štampača „MPS 1200" pokazuje da se u strategiji firme ipak nešto menja.
Više ne morate da tražite bolji štampač za vaš „komodor", jer će temperamentni mališa veoma dobro obavljati svoj posao i neće vam praviti probleme sa povezivanjem. Dobro je i to što za ovaj štampač nećete morati da kupujete „komodor", jer će sasvim dobro raditi u sprezi sa bilo kojim drugim računarom, što baš i nije bila odlika prethodnih „komodorovih" modela.

Razmerno loš kvalitet otiska kod štampača serije „800" podsticali su vlasnike da traže i drugačija rešenje. Sve bi bilo u redu da „komodor 64" nema nestandardne interfejse koji su perfekcioniste dovodili do očajanja.
Da bi „komodor" zadržao epitet velike i pouzdane firme morao je da se potrudi da popuni rupe u svom proizvodnom programu. Tako se pojavila nova serija štampača od kojih vam predstavljamo model „MPS 1200".
Nesrazmerno velika kutija krije kompaktan štampač sa svim dodatnim delovima i uputstvom za upotrebu. Dobili smo serijski interfejs koji je prilagođen „komodoru 64" i posle nekoliko bezuspešnih pokušaja da štampač spojimo sa redakcijskim „timom" odustajemo. Štampač se uopšte ne odaziva. Serijski interfejs je namenjen isključivo za povezivanje sa „komodorom 64" i u tom slučaju povezivanje je veoma jednostavno.

Tehničke karakteristike štampača

Commodore MPS 1200

Mehanika
Dvosmerni matrični štampač
9-pinska glava za štampanje
Brzina
Deklarisana
120 CPS Draft i 24 CPS NLQ
Izmerena
79 CPS Draft i 18.5 CPS NLQ
Matrice
Standard 9×9
NLQ 17×17
Podebljeno 9×9
Uvećano 9×1D
podebljeno i
uvećano 9×19
Tipovi znakova
Pica 10 CPI
Elite 12 CPI
Sabijeni 17 CPI (pica)
2D CPI (elite)
Rešireni, podebljeni, masni, indeksni, stepeni, inverzni, proporcionalno razmicanje

Razmak između znakova
5, 6, 8.5, 10, 12, 17 ili 20 CPI
Razmak između linije
Standardni 1/6, 1/8 ili 7/72 inča
Programabilni n/72, n/144, n/216
Papir perforirani u rolni ili pojedinačni listovi A4 formata

```
Racunari 38 - test stampaca
Commodore MPS 1200

NORMAL DRAFT:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

ITALIC DRAFT:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

NLQ NORMAL:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

NLQ ITALIC:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

BOLD NLQ:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

UNDERLINED NLQ:
1234567890 .,:;+-*/?()^@#$
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ

INVERTED:
1234567890 .,:;+-*/?()^@#$
[illegible]
ABCDEFGHIJKLMNOPQRSTUVWXYZ

H2SO4       SUBSCRIPT
E=mc2       SUPERSCRIPT

Racunari
Racunari
Racunari
Racunari
Racunari... Racunari
```

Nije potrebno nikakvo podešavanje interfejsa i printer podržava sve „komodorove" znakove i ispisne modove. Ako želite da vam se štampač odaziva u „epson FX" kompatibilnom modu morate da izvadite interfejs iz ležišta i potražite DIP prekidače kojima možete da podesite karakteristike štampača odmah po uključenju. Tu je i komplet prekidača za podešavanje nacionalnih setova znakova.
Teškoće sa povezivanjem „time" su nastale verovatno zbog prirode „komodorovog" serijskog interfejsa koji je pravljen sa „open colector" TTL kolima i zahteva dovođenje napona na odgovarajuće linije za povezivanje. Povezivanje sa standardnim RS 232C interfejsom bi verovatno moglo da se izvede uz malo muke ali mi smo se odlučili za praktičnije rešenje. Sa „komodorom 64" interfejs naravno radi savršeno. Pozvali smo prodavnicu „Metalservisa" koja prodaje „Konimove" uvozne artikle i zatražili paralelni interfejs. Rešenje se pokazalo kao veoma uspešno pa ga preporučujemo vlasnicima „amige" i „nekomodorovskih" računara.
Kakve zamerke bismo mogli da uputimo štampaču? Pre svega da je veoma „plastičan" i izgleda vrlo krhko (posebno se odnosi na traktor za perforirani papir). Plastični je i remen koji vuče glavu štampača preko papira što bi kod duže upotrebe moglo da pravi probleme. Problem ne predstavlja problem ukoliko je uvoznik u stanju da obezbedi novi remen ako vam se ovaj istegne. Treća zamerka bi se odnosila na nedostatak hladnjaka na glavi štampača.
Dobar deo naših strahova je nestao tokom upotrebe printera. I pored krhkog izgleda štampač je pokazao izdržljivost koju treba pohvaliti. Glava se prilično precizno pomerala i nije se vidno grejala uprkos činjenici da je prevalila mnogo šlajfni koje čitate u ovom broju. Štampač je položio test izdržljivosti ali su ostale neke zamerke pri upotrebi pojedinačnih listova papira. Hartija ponekad zapne za glavu pa je potrebno prethodno napraviti „uši" pre ubacivanja lista, to jest zavrnuti hartiju na rubovima. Ni detektor kraja papira se nije baš iskazao pa štampač ponekad nastavi da piše jedan red preko drugog u dnu strane.
Uprkos svim zamerkama koje nisu velike i koje se mogu otkloniti uz malo iskustva u radu dobija se kvalitetan i kompaktan štampač sa dobrim ispisom. Najvažniji je ipak krajnji rezultat koji zaslužuje pohvale. Deo raznovrsnosti ispisa štampača je prikazan u prilogu teksta. Uz nisku cenu i pristojne karakteristike ovaj štampač će vam pružiti mnogo zadovoljstva u korišćenju.

Voja Gašić

Adresa zastupnika: KONIM, Ljubljana, Titova 38, telefon 061/312-290

Zvezdane staze

Izuzetno retko nam se ukazuje prilika da na ovim stranicama prikažemo i po neki proizvod iz domaće softverske radionice. To, međutim, izgleda kao da je garancija za kvalitet.
Program koji želimo ovom prilikom da predstavimo je PC SKY ATLAS dr Dušana Slavića i Ninoslava Čabrića. O autorima ne treba mnogo govoriti, pošto ih naši čitaoci već dobro poznaju, ali o programu ima šta da se kaže, jer on već neko vreme izaziva pažnju u javnosti.

Da li znate grčki mit o Perseju i Andromedi? Ako ne znate mogli bismo ukratko da vam ga prepričamo.

Etiopijom je kažu nekada davno vladao car Cefej Njegova žena carica Kasiopeja se, kažu i to, stalno hvalisala svojom lepotom što je na kraju dojadilo izvesnim nereidama pa su tužile Kasiopeju bogu mora Posejdonu Ovaj je da bi umirio pomenute nereide poslao na obale Etiopije nekakvo grozno morsko čudovište Siroti car Cefej je da bi umilostivio gnev bogova morao da žrtvuje svoju kćer Andromedu vezavši je za stenu i ponudivši je čudovištu Ali jasno tu je usput prolazio poznati junak Persej jašući na krilatom konju Pegazu On odluči da ubije čudovište i oslobodi Andromedu na sveopšte zadovoljstvo

Možda pomalo liči na Dinastiju ali mi tu ne možemo da se mešamo Ispričali smo vam ovu priču samo zato da bismo nekako skrenuli razgovor na astronomiju

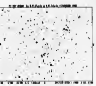
Kakve veze ima Etiopija jedno čudovište i jedan krilati konj sa astronomijom? Pa baš nikakve Ali gotovo svi likovi iz priče danas se nalaze na nebu u vidu sazvežđa Naravno ne očekujte da kada vam neki astronom pokaže sazvežđe Andromeda ugledate tamo na nebu prekrasnu gorepomenutu kćer cara Cefeja Sve je to stvar poprilične mašte, a astronomi toga imaju na pretek Sazvežđa i grčki mitovi samo olakšavaju snalaženje na nebu

Danas imamo tačno 88 sazvežđa Često se čuje i mišljenje da ih ima 89 jer se jedno od njih (Zmija) uslovno deli na dva (Glava zmije i Rep zmije)

Ukupan broj zvezda koje vidimo golim okom u svim tim sazvežđima iznosi oko 6000 Postoje *zvezdani katalozi* koji za svaku od tih zvezda (pa i za mnoge od onih koje se vide samo teleskopom) daju tzv *ekvatorske koordinate* (rektascenzija i deklinacija) čime je njihov položaj na nebu potpuno zadat

Sunce Mesec i planete se kreću među tim zvezdama na nebu i njihove koordinate se neprekidno menjaju pa se ne daju u katalogu Moraju se ili računati ili uzimati iz posebnih tablica *(efemeride)*

Poseban problem je ako želimo da saznamo da li se neko nebesko telo u nekom trenutku vidi iz nekog mesta na Zemlji i ako se vidi u kom delu neba ga treba tražiti Razume se, svaki student astronomije će vam brzo pokazati formule po kojima se takav račun obavlja nekoliko sinusa i kosinusa par transformacija vremenskih trenutaka i — stvar je gotova Ali ako treba sve to ponoviti više puta teško da ćete naći



nekog ko bi se ponudio da to dobrovoljno uradi

Računanje je, naravno pravi posao za računar Ukucate ulazne podatke a mašina vam izbaci rezultate U memoriji računara mogao bi čak biti sadržan i čitav zvezdani katalog a efemeride tela Sunčevog sistema mogu se računati za bilo koji trenutak po poznatim formulama teorijske astronomije Upravo sa tom idejom urađen je i program PC SKY ATLAS

Crtani film

Program raspolaže katalogom sa oko 8500 zvezda i posebnom rutinom za računanje koordinata Sunca Meseca i planeta Za date ulazne podatke (vreme datum nebeske koordinate geografske koordinate itd) na ekranu ili štampaču se iscrtava izgled neba dopunjen osnovnim krugovima nebeske sfere (meridijan ekvator ekliptika horizont galaktički ekvator) Zvezde su prikazane kao tačke manjeg ili većeg prečnika zavisno od sjaja, Sunce i Mesec kao nešto veći kružići (Mesec u odgovarajućoj fazi) dok su planete date uobičajenim astronomskim oznakama

Slike se mogu snimiti na disk i kasnije unositi uz mogućnost *sukcesivnog prikazivanja* (animiranja) što je izuzetno podesno za filmsko simuliranje neke pojave

Program je pisan za PC XT ili AT sa *herkules* grafičkom karticom i matematičkim koprocesorom Hard disk je poželjan za čuvanje sukcesivnih slika i njihovu kasniju reprodukciju

Minimalna memorija potrebna za rad je 512K od čega se jedan deo konfiguriše kao RAM disk To izuzetno ubrzava rad programa jer je komunikacija sa diskom veoma zastupljena (naizmenično se pozivaju pojedini moduli i datoteke)

Program je pisan u paskalu i fortranu a veza između modula ostvarena je preko BAT datoteka u okviru DOS-a

„Turbo planete

Turbo paskal (verzija 3) korišćen je za deo programa koji računa koordinate Sunca Meseca i planeta u dvostrukoj tačnosti To je zapravo program Ninoslava Čabrića razvijen još 1962 za računar šarp 80K i kasnije prilagođavan drugim računarima, a na osnovu koga se štampaju efemeride u časopisu „Vasiona

Osim toga u paskalu su pisane i sve rutine za komunikaciju sa korisnikom Iz razgovora sa Ninoslavom Čabrićem saznali smo da program retko gde koristi standardne procedure **WRITELN** i **READLN**, zbog njihove ekstremne sporosti i nekomfornosti Umesto toga razvijene su veoma brze posebne procedure za ispis, koje direktno adresiraju ekran Efekat je zaista impresivan Sadržaj ekrana sa tekstom pojavljuje se praktično trenutno

Matematički koprocesor znatno ubrzava račun koordinata Dovoljno je reći samo to da se položaj svih planeta Sunca i Meseca (sa fazom Meseca) dobijaju za vreme manje od jedne sekunde Bez koprocesora to bi išlo i na pola minuta ,

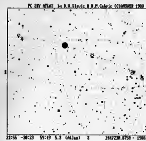
Autori su međutim svesni da je koprocesor na ovim našim prostorima prilično redak luksuz pa postoji ideja da se uradi i verzija koja će raditi baz koprocesora Jeste da će biti sporo ali bar će biti dostupno širem krugu korisnika

Celobrojni račun

Drugi deo programa uzima koordinate zvezda iz kataloga kao i izračunate položaje tela Sunčevog sistema i vrši projektovanja (grafički prikaz) neba na ekranu Pisan je u fortranu *MS F—77* verzija 4.01 ali ta činjenica sama po sebi nije dovoljna da bi se postigla fascinantna brzina projektovanja kojom radi PC SKY ATLAS Verovali ili ne proračun koordinata svih 8500 zvezda zajedno sa krugovima nebeske sfere gotov je za jednu ili dve sekunde Jasno je da za to nije zaslužan ni kompajler ni koprocesor već — programer

Kako nam je prof Dušan Slavič objasnio kompletan proračun grafičkog prikazivanja zvezdanog neba obavlja se u celobrojnoj aritmetici korišćenjem bogatih mogućnosti fortrana 4.01 Izbegnuti su pozitivi trigonometrijskih funkcija a umesto toga vrednosti se uzimaju neposredno iz tabela Jseno u čitav program ugrađeno je višego-

330



dišnje iskustvo prof. Slavića na polju numeričkih metoda i računanja elementarnih funkcija. Dovoljno je da se prisetimo serije članaka. To može i bolje."

Razume se, na ekranu se ne vide uvek svih 8500 zvezda. Ali program mora da ispita svaku, da bi utvrdio njen položaj na projekciji. Mnogo se uštedelo na brzini tako što se prvo testira udaljenost zvezda od centra slike, pa se samo u slučaju da zvezda pada na ekran računaju ravanske koordinate.

Na vremenu se štedelo gde god i koliko god je to bilo moguće. Autori kažu da je od prvih verzija programa do danas brzina porasla i do *deset puta*.

Interesantno je napomenuti koliko se problema javljalo u radu zbog *bagova u fortran prevodiocu*(!) O tome bi se mogao napisati poseban tekst u „Računarima", koji bi verovatno dobro došao svim programerima. Recimo, kompajler ima čudnu osobinu da, prilikom optimizacije koda, jednostavno *ukloni* neke promenljive i tako učini da program krahira. Programer mora da se dovija na razne načine kako bi postigao ono što želi.

Izuzimajući bagove, jedan od najvećih programerskih problema bio je prikazivanje zvezda. Jasno je da sjajnijoj zvezdi treba da odgovara veći kružić, ali *po kojoj skali* to raditi? Ako su kružići preveliki, ekran će biti pretrpan i nepregledan. Ako su premali, sazvežđa neće biti prepoznatljiva. Prof Slavić kaže da je utrošio barem 1600 listova za razne probe grafika. Rezultat tog truda može se videti iz priloženih slika, koja deluju kao da su preuzete iz zvezdanog atlasa.

Prijateljski program

Autori su se maksimalno trudili da program učine pristupačnim i onima koji baš nisu profesionalni astronomi. Sve svoje želje korisnik unosi preko *glavnog menija*, gde se ukucava vreme, datum, ekvatorske koordinate željene tačke na nebu, geografske koordinate tačke posmatranja na Zemlji, veličina slike i granična magnituda zvezda. Pritiskom na neki od funkcijskih tastera poziva se zatim željena opcija.

U režimu *ATLAS* iscrtava se karta neba sa centrom u zadatoj tački i sa zadatim prečnikom. Projekcija je *polarna ekvidistantna* (ekvivalentna kartama zemljinih ili nebeskih polova). Ukoliko to zahtevamo, program će iscrtati i projekcije svih važnijih krugova nebeske sfere.

U režimu *SKY* slika se crta sa centrom u zenitu mesta posmatranja u istoj projekciji. Sa takve slike se vrlo lako može pročitati koja nebeska tela su u datom trenutku iznad, a koja ispod horizonta.

Izbor koordinate tačke na nebu može se ostvariti i posredno, zadavanjem neke zvezde ili čitavog sazvežđa. Program raspolaže datotekom sa položajima 102 najsjajnija zvezde kao i položajima i veličinama svih sazvežđa.

Koordinate mesta na zemlji se slično tome mogu zadati izborom grada iz datoteke od 100 najvećih gradova na Zemlji, plus severni i južni pol kao posebne posmatračke stanice.

Osim toga, svaka nova slika može biti formirana na bazi prethodne pomeranjem krstića — kursora (izbor centra slike) i zumiranjem (izbor prečnika slike).

I naravno, najinteresantniji efekat je *animacija*, odnosno sukcesivni prikaz više slika. Slike se prikazuju u rasponu od dve sekunde, čime se može simulirati tok neke astronomske pojave, kao što je kretanje Sunca, Meseca ili planeta među zvezdama. Svaka slika zauzima na disku 32K (herkules format). Hard disk može tu značajno da uveća mogućnosti programa.

Ne računajući prethodni rad i iskustvo, samo na ovom programu autori su radili skoro pola godine. Učinjeni su koraci da se program plasira na zapadno tržište, a kako nam je rečeno, biće postavljen uslov da strani partner besplatno ustupi pravo domaćem izdavaču da distribuira program u našoj zemlji. To je lep gest jer predstavlja praktično jedini način da program bude pristupačan i domaćim ljubiteljima računara i astronomije.

Jovan Skuljan

Telekomunikacioni eksperiment

Aprilske bubice

Slušaoci „Programa u boji" Radio-Beograda i hakeri koji su detaljno proučili listinge objavljene u prošlim „Računarima" svakako znaju da je naš „Telekomunikacioni eksperiment" predstavljao samo računarski uobličenu prvoaprilsku šalu. Prava šteta — zar ne bi bilo lepo kada bi radio talasi mogli da prenose podatke koji bi bili nečujni za ljudsko uho i razumljivi za mikroprocesore!

Mala ekipa delegirana od strane redakcije „Računara" učestvovala je u 31-martovskom „Programu u boji" koji je šarmantno vodio čitaocima „Galaksije" dobro poznati *Vladimir Jelenković*, urednik Naučnog programa TV Beograd. „Eksperiment" je počeo četrdesetak minuta pre ponoći emitovanjem programa za spektrum, komodor 64, amstrad i BBC (program za IBM PC smo izostavili i iz „Računara", zaključivši da i onako ne bismo mogli da ga emitujemo preko radio talasa) — posle gotovo četiri godine Radio Beograd je ponovo emitovao kompjuterske programe. Tačno u 23 časa i 30 minuta radio je počeo da emituje signale nečujne za ljudsko uho, koji su upravo takvi i bili — niko ih nije čuo jer je emitovana jedino dobra muzika! U istom trenutku mnogobrojni čitaoci „Računara" su startovali programe koji su ispisali nešto poput „Izvršavanje u toku" i vrteli se u (skoro) mrtvoj petlji čekajući prvi april.

U međuvremenu se u studiju razgovaralo o računarima, interakciji sa radio signalima, američkim FCC propisima, provalama u velike kompjuterske sisteme i sličnim pikanterijama. Telefon je neprekidno zvonio, uglavnom zahvaljujući činjenici da su programu za Komodor 64 iz prošlih „Računara" nedostajali neki bajtovi — komodoristi koji nisu snimili program bili su očajni što ne mogu da se uključe u eksperiment.

Tačno minut posle ponoći računari naših čitalaca koji su dosledno sledili uputstva ispisali su „Aprilllll" i sve je postalo jasno. Telefoni su prirodno i dalje zvonili — vezu je najpre dobio *Nebojša Dimitrijević* iz Svetozareva koji nam je čestitao na duhovitosti i zaslužio nagradu — jednogodišnju pretplatu na „Računare" (onih 100.000 dinara i onaj računar koji je pominjao Vlada Jelenković bili su, naravno, samo deo aprilske šale). Simpatije svih učesnika emisije i, verujemo, svih slušalaca radija osvojio je trinaestogodišnji *Bojan Grubač* iz Beograda koji je priznao da je u toku emisije malo zadremao ali se probudio na vreme da pročita pozdravnu prvoaprilsku poruku. Telefoni su nastavili da zvone kada smo, jedan čas posle ponoći, napustili zgradu Radija prepuštajući neumorni „Program u boji" drugim učesnicima.

Veoma blamo se razočarali da brojni čitaoci „Računara" nisu unapred shvatili u čemu je štos — programi su bili dovoljno kratki da ih je čak i površni poznavalac mašinca mogao analizirati. Svi koji su se susreli sa zaštićenim igrama svakako znaju da se segmenti programa mogu gotovo neograničeno zakomplikovati ali se oni time i produžavaju — naši programi su morali da budu kratki da bi se čitalo ko uopšte upustio u njihovo kucanje! Zato je primenjena zaštita bila rudimentarna — poruka Aprillllh je bila XOR-ovana nekom konstantom kako je običan ASCII *dump* ne bi pronašao!

Prvo pismo poslao nam je *Dejan Milenović* iz Niša koji je, premda mu se tekst učinio savršeno logičnim, posumnjao u onaj 31. mart i isprobao program mnogo ranije. *Dejan* nas je uzgred budi rečeno na kraju pisma proglasio za „najduhovitiji od svih časopisa koji se bave popularizacijom računarstva" i poželeo da šala što bolje uspe. Slično je reagovao i *Branislav Tomanović* iz Doboja koji kaže da „malo zafrkancije nije na odmet ni u tako ozbiljnoj rubrici kao što su AKCIJE — svi mi imamo dovoljno problema sa našom situacijom pa malo smeha dobro dođe. A najbolje je smejati se na vlastiti račun". Jedva par dana kasnije primili smo i pismo *Ivana Jovanovića* iz Niša koji kaže da je eksperiment bio „još jedan dokaz kvaliteta i originalnosti „Računara" koji su bili i ostaju superlativ za mnoge druge". „Budite sigurni, hakeri su uz vas", završava pismo drug Jovanović a mi možemo samo da odgovorimo: Budite sigurni, i mi smo uz njih. Većina ostalih pisama (prvo je poslao *Dragan Vasić* iz Niša) sadržalo je samo jednu reč — Aprillll!

Par dana pre zaključivanja ovih „Računara" urednik „Radio Beograda" i naših „Razbarušenih sprajtova" *Slobodan Perović — Mutko* upoznao nas je sa zvaničnim demaršom koji je „Radio Beograd" primio od računskog centra jedne poznate banke koju ovde nećemo imenovati. Odgovorne organe centra sablaznula je čak i sama pomisao da će radio emitovati signale koji mogu da oštete važne i poverljive podatke upisane na računarske diskove!

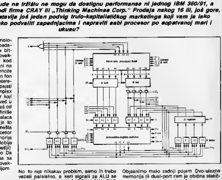
I tako će po uspešnom završetku „Telekomunikacionog eksperimenta" naša rubrika Akcije ponovo postati i ostati ozbiljna, bar do nekog budućeg prvog aprila!

Dejan Ristanović

Kako sklopiti procesor

Procesori koji se danas nude na tržištu ne mogu da dostignu performanse ni jednog IBM 360/91, a kamoli mašina kakve proizvodi firma CRAY ili „Thinking Machines Corp." Prodaja nakog 16 ili, još gore, 32 bitnog procesora predstavlja još jedan podvig trulo-kapitalističkog marketinga koji vam je lako izvukao pare iz džepa. Kako podvaliti zapadnjacima i napraviti sebi procesor po sopstvenoj meri i ukusu?

Jedna solucija je da se od GyAs tehnologije napravi pravi procesor sa svim pripadajućim delovima a druga je upotrebe bit-slajsa (nezahvalno je davati prevod za ovakvo ime) Suština je u tome što su kod bit-slajsa svi delovi procesora odvojeni na različite čipove Prigodnim spajanjem može se dobiti normalni procesor prilagođen fon Nojmenovoj arhitekturi ali to nije interesantno Mnogo je bolje te delove spajati tako da se dobije CISC procesor (Complex Instruction Set Computer — procesor koji ima mnogo složenih naredbi koje su već u njemu — obrnuto od RISC) koji ima keš od 16 K i koma korisnik može da definiše mikroinstrukcije Verovatno većina čitalaca zna šta je keš ipak ponovimo da je to jedan bafer u koji sistem unapred smešta podatke iz spoljašnje memorije da procesor ne bi čekao na njih (dok se memorija spremi da odgovori procesoru, on dobija proširenje vena od stajanja u mrtvoj petlji) Sve te karakteristike zvuče veoma lepo Da bismo to i dokazali upoznaćemo se sa danasnajrasprostrajenijom familijom ovakvih čipova takozvanom AMD 2900 serijom



Procesor na delove

Ako se zapitate šta je sve potrebno za jedan procesor videćete da spisak i nije baš toliko dugačak Potrebna je ALU (Arithmetic Logic Unit — aritmetičko-logička jedinica) registri dober šifter (šift registar) dakoder mikroinstrukcija i memorija za mikroinstrukcije Mnogi će se pitati šta je sa esencijalnim delom svakog procesora — programskim brojačem Ništa — on nije potreban! Adresu sledeće instrukcije možemo smestiti u samu instrukciju i tako dobiti program bez skokova ili poziva potprograma Listinzi takvih programa nisu lepi ali je glavni uzrok svih patnji otmenih programera — instrukcija GOTO — iskorenjena u potpunosti Ako neko voli struktuirano programiranje može programski brojač dodati veoma jednostavno

To je bio spisak glavnih delova Da vidimo šta nam može ponuditi 2900 serija Čip pod nazivom 2903 je tzv microprocessor slice Najjednostavniji način da se utvrdi šta ovaj čip radi je da se zaviri u sliku 1 Kao što vidimo to je skup izvršnih delova procesora U njemu su integrisani ALU šifter kao i registarska matrica od 16 univerzalnih registara Svi potrebni signali su izvedeni na samom čipu pa tako možete praviti pravi RISC bez dekodiranja instrukcija (nevolja je u tome što će vam biti potrebno 60-tak ako ne i više data linija) Još jedna osobenost ove serije čipovi rade sa samo 4-bitnim podacima (nibiovima)

No to nije nikakav problem, samo ih treba vezati paralelno, a keri signali za ALU se generišu preko 2902 CARRY LOKAHEAD generatora Sta više na čipu postoje ulazi za definisanje važnosti' čipa (signali MSS i LSS) Tu je i neizbežni CLK (odnosno kako je ovde nazvan CP) ulaz tek toliko da znate gde da spojite oscilator pravougaonih impulsa Njegova širina može da bude od 80 ns (to je 12 5 MHz) pa nadalje (radi usporedbe — 68020 do 25 MHz do paradoksa da je bit slajs sporiji od novih procesora Da li je tako? Ako koristimo standardne memorije sa vremenom pristupa od 100 nS MC68020 na 25 MHz (40 nS) će imati tek nešto malo bolje vreme od 68000 na 10 MHz (gledajući neki realan program) To je zbog toga što programerima uvek nedostaje samo još jedan registar pa moraju da koriste sporu memoriju Uz to, instrukcije se same po sebi, moraju uzimati iz memorije Faktički gorepomenuta dvadesetica će biti duplo brža od 68 K samo za vreme stektivnog izvršavanja instrukcija i tu vam možemo popraviti raspoloženje time što ćemo reći da bit-slajs koristi jedan dober štos za skraćivanje svog radnog vremena ali o njemu malo kasnije

Kao što se sa slike vidi, ulazi ALU se mogu birati da budu izlazi registara ili data ulazi izlaz ALU može biti u pomerački (shift) registar (Q'' registar) i/ili na data linije Registarska matrica je u stvari dual-port ram koji nezavisno može primati podatke u dva registra, čime se znatno skraćuje vreme izvršavanja nekog programa

Objasnimo malo zadnji pojam Dvo-ulazna memorija ili dual-port ram je obična statička memorija koja ima interfejse za povezivanje na dva različite strane Znači ima dve adresne sabirnice kao i dve sabirnice za podatke Međutim, samo je jednoj grupi sabirnica dozvoljeno da vrše kontrolu upisa u memoriju Princip rada možemo pojednostaviti i reći da se podaci iz dve različite memorije prepisuju za vreme kada ih spoljašnji svet ostavi na miru odnosno kada nisu selektovane Šta to znači za microprocesor-slajs? To je veoma bitna stavka — zamislite da morate da saberete sadržaje dva registra U normalnim procesorima (čitaj gore pomenuta 68020 na 25 MHz!) to zahteva 4 instrukcijska takta čitanje prvog registra čitanje drugog registra sabiranje i na kraju upisivanje u registar Pri upotrebi dvo-ulazne memorije za registre, ova operacija se izvršava u 2 takta pribavljanje operanada i sabiranje je u jednom taktu, a drugi je zadužen da se podaci upišu u odredišni registar Pa zašto onda svi ne koriste dual-port ram za registre? iz jednostavnog razloga — na čipu nema mesta! (Svi znaju da su sve dobre stvari ili nemoralne ili zabranjene ili debljaju — u ovom slučaju silicijumski čip) Kod većih sistema kojima je procesor cela ploča, ovakav sistem se neizostavno ugrađuje

Signali SIO i QIO su namenjeni prenosu signala prilikom šiftovanja podataka koji su duži od 4 bita ALU ima 9 instrukcijskih ulaza (to ne znači da ima i 512 funkcija —još jedna podvala sunčane Kalifornije) Svi

ostali neophodni podaci se vide sa slike a detaljnije informacija možete dobiti od proizvođača

Mikroinstrukcijski sekvencer

Drugi važan deo je mikroinstrukcijski sekvencer Njegova uloga je da kada primi instrukciju iz programske memorije pokrene odgovarajući program koji upravlja radom izvršnih delova procesora Sekvenca tih instrukcija se prirodno zove mikroinstrukcija a dobija se iz tzv mikroinstrukcijske memorije Ceo sekvencer se sastoji od već pomenute mikroinstrukcijska memorije (njeni izlazi za podatke se vezuju na kontrolne ulaze microprocesor-slajsa) kao i jednog malog DMA sklopa koji za svaku određenu instrukciju (koja predstavlja adresu za memoriju) generiše n dodatnih adresa nižeg značaja Princip je isti kao i kod karakter generatora na „galaksiji" Sekvencer je u starijim udžbenicima (u našem slučaju to znači u udžbenicima) koji se bave naukom o kompjuterima onaj deo računara obeležen kao „Upravljačka jedinica" Još je jedna bitna stavka vezana za ovaj deo — mnogi tvrde da je on usporivač procesora Po autorovom mišljenju to uopšte nije tačno — korisnik je najsporiji, a tu mu se približava i ukupna memorija računara (ne samo RAM/ROM nego i diskovi) Svejedno ako je potrebno podeliti dva broja u pokretnom zarezu od 80 bita pitanje je samo da li ćemo taj program mi praviti ili su robovi sa zapada to učinili umesto nas i to zaračunali u cenu Ako ga sami pravite, računajte i na dodatne memorijske cikluse za dobavljanje instrukcije iz memorije

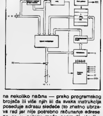
Prilikom objašnjavanja funkcije sekvencera namerno nije pomenuto koja to memorija treba da se ugradi da bi sistem radio Naravno, to može biti PROM ali i EEROM Onda dobijamo interesantne mogućnosti — korisnik može programirati dejstvo svakog koda, mašinskog jezika na procesor Ako se pitate zašto EEROM, a ne RAM, zamislite se kako ćete uključiti mašinu ...

Pređimo na konkretne stvari i pogledajmo sliku 2 Na njoj je dat sekvencer 2910 Ako ga malo bolje pogledate videćete da je tu i nekakav stek Istina, kapacitet mu je samo 5 reči, ali ako ne može bolje ...
Ti ulazi se spajaju na memoriju sekvencera tako da ona kontroliše i ovaj blok. Naravno, i ovde je prisutan CLK ulaz (on se ovde zove CL), i to je druga tačka gde bi trebalo da spojite onaj malopre pomenuti oscilator Ne morate ga menjati — i ovde je maksimalna brzina 80 ns I za kraj opisa jedna sitnica — sama memorija sekvencera mora biti malo brže od njega samog

Veza sa memorijom

Ostalo je još pitanje kako spojiti procesor sa spoljašnjim svetom Pošto po definiciji nemamo dovoljno para od interesa je sastaviti interfejs koji će biti što jednostavniji i koji će omogućavati priključenje što jevtinije memorije Prvo što treba da raščistimo to je koliko će naš računar imati nivoa memorije — dva ili tri (dva nivoa memorije obuhvataju RAM i spoljašnju memoriju a tri nivoa sve to plus keš) Ako planiramo i keš to je već krupan zalogaj i nećemo se upuštati u takve rasprave (da imate i vi o čemu da razmišljate ...) Ako ga ne planiramo, memorija se može povezati





na nekoliko načina — preko programskog brojača ili više njih ili da svaka instrukcija poseduje adresu sledeće (to znatno ubrzava rad jer nije potrebno računanje adrese pa se u potezu može napraviti pipeline arhitekture) U tom slučaju možete podvaliti piscima sistemskog softvera tako što ćete jednu liniju instrukcijske magistrale proglasiti za R/W signal — prilikom svake instrukcije moraju obraćati pažnju da li se nešto čita ili upisuje u memoriju Pitanje je i kakve memorije koristimo — statičke ili dinamičke Ako koristimo one jeftinije moraćemo da obezbedimo i osvežavanje ali to nije problem ako u računaru imamo DMA (a šta je računar bez DMA — to ima čak i „spektrum") Odgovore na sva ova pitanja moramo sami naći, pošto Advanced Micro Devices (firma koja proizvodi 2900 seriju) ne pravi čipove za gradnju MMU (dela procesora koji se bavi isključivo upravljanjem spoljašnjom memorijom) Ali ako smo napravili procesor ovaj interfejs i nije neki problem

Načini spajanja

Opisali smo dve vitalne jedinice svakog procesora pa sada prelazimo na to kako ih možemo spojiti Nije interesantno razmatrati kako dobiti procesor 2 puta bolji od 68 K nego procesor koji je 20 puta brži Možda zvuči malo ambiciozno ali i to je moguće

Ako nastavimo da razmišljamo kao i Janoš fon Nojman nećemo daleko stići Maksimalna propusnost ovih čipova sa tom arhitekturom je oko 12.5 MIPS (bez obzira na dužinu osnovne reči), ali je to suva teorija koja naravno ne radi Potrebno je nešto bolje!

To bolje se može postići na nekoliko načina Prvi je da umesto jedne ALU konstruišemo odmah više njih i tako dobijemo primitivan oblik paralelizma (takođe je potrebno i povećati broj unutrašnjih sabirnica) Druga stvar je razdvajanje instrukcijske i magistrale podataka Samo ovim zadnjim poboljšanjem dobijamo 30—70% brzine Isto tako možemo uvesti protočnu arhitekturu (kada je ima i CRAY 1 onda možemo i mi) Ne zaboravimo dobri stari keš (ima ga IBM 360 — nisu sve to otkrića našeg doba) No onda otpada interesantno preskakanje programskog brojača Možemo dogrediti sistem tako da on više ne obrađuje skalare nego vektore i tako dobiti vektorski procesor Objasnimo i ovaj pojam Kao što znamo normalni procesori rade samo sa pojedinačnim podacima koje u matematici nazivamo skalari Matematika nam isto tako kaže da više podataka možemo predstavljati kao niz skalara i tako dobijamo vektore Vektorski procesori normalno operišu sa vektorima ali i sa skalarima Štos je u tome da svaki registar ovakvog procesora ne sadrži jedan broj, nego niz brojeva Kada sabiramo dva registra sabiramo prvi podatak jednog operanda sa prvim podatkom drugog operanda itd Ceo sistem liči na paralelno sabiranje više skalara, što u stvari i jeste Naše je samo da odredimo dimenziju vektora sa kojim ćemo raditi tj da ustanovimo broj skalara koji će predstavljati taj vektor (na žalost ovaj broj ne može težiti beskonačnosti jer je za povećavanje dimenzije vektora za jedan potrebno dodati još jednu ALU jednu koordinatu u registarskoj matrici itd) Kao što se zna iz matematike, postoje operacije koje dva vektora pretvaraju u samo jedan skalar A šta da radimo sa skalarom kada on nije osnovni tip podataka? Zbog toga se i moraju predvideti neke operacije sa skalarima kao što je ubacivanje skalara na određeno mesto u vektoru i slično

I na kraju, možemo kodirati veoma složena instrukcije optimizovati ih i tako dobiti procesor koji direktno podržava neki proceduralni jezik visokog nivoa Iako ovo nije direktna ušteda u brzini setimo se da se mnogo više vremena troši na pisanje kompajlera i sličnih alatki nego na samo izvršavanje programa (ovo ne važi za korisnička programe ali ko je video da se radi sa dBASE na TI ASC mašini ili STARAN-u) Ako sve to nije dovoljno, setite se da uvek možete duplirati broj bitova u reči sa kojom radite Bit-slajs to velikodušno omogućava

Pošto smo sve ovo ovako lepo izložili, postavlja se logično pitanje — zašto svi procesori nisu urađeni u tehnologiji bit-slajsa? Odgovor je — jesu! Upravo su procesori na većim sistemima takvog tipa Međutim, bit-slajs se najviše koristi pri razvoju prototipa procesora pa se takav procesor kasnije integriše na samo jedan čip (i time se na žalost izgubi dosta lepih stvari)

Ako želite da pravite procesor detaljnije informacije možete dobiti kod Advanced Micro Devices 901 Thomson Place Sunnyvale California 94086

Srđan Mijanović

Intel 80387

Treća generacija

Aritmetički koprocesori za IBM PC ne predstavljaju posebnu novost za čitaoce „Računara" — prilično smo ih detaljno predstavili u „Računarima 29" i „Računarima 35". U međuvremenu je Intel lansirao treću generaciju svojih aritmetičkih koprocesora koju predstavlja 80387 — parnjak 32-bitnog mikroprocesora 80386. Preuzimajući podatke iz napisa „Programming the 80387 Coprocessor" koji je objavljen u martovskom „Bajtu", pokušaćemo da i našim čitaocima predstavimo novitete koje 80387 donosi.

Iako svi znamo da aritmetički koprocesor dramatično ubrzava neke aplikacije, relativno se mali broj korisnika odlučuje za njegovu nabavku. Razloge, pre svega, treba tražiti u ceni koja je bila prevelika čak i kada se radilo o „običnom" koprocesoru 8087: vredi li uložiti 100 dolara u čip bez koga računar savršeno radi? Odluku je još teže doneti kada se radi o AT-u ili računarima opremljenim mikroprocesorom 80386: 80287 trenutno košta 300, a 80387 čitavih 500 dolara! Pre bacanja u ovakav trošak svakako treba proučiti performanse čipa koji se dobija.

Arhitektura

Procesor 80387 je specijalni 80-bitni koprocesor koga proizvodi isključivo Intel (u stranoj literaturi ćete pronaći termin *solesource* koji znači da neka firma ima monopol na proizvodnju nekog čipa). Sa slike 1 vidimo da je 80387 opremljen sa 8 80-bitnih, dva 48-bitna i tri 16-bitna registra u koje se pod kontrolom mikroprocesora 80386 upisuju racionalni brojevi. Koprocesor ove brojeve može da sabira, oduzima, množi, deli, korenuje i poredi pri čemu se svaka od ovih operacija obavlja stotinak puta brže od odgovarajućeg solidno napisanog mašinskog programa za cenovni mikroprocesor. Za razliku od svojih prethodnika 80387, može da računa i vrednosti elementarnih trigonometrijskih funkcija sinus i kosinus, dok ga za računanje ostalih transcedentnih funkcija treba programirati! Čak i ovakvo programirano izvršavanje je, međutim, bar pedeset puta brže od programa koje izvršava cenovni mikroprocesor.

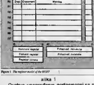


Figure 1 The register model of the 80387

slika 1

Ovakvo unapređenje performansi se poštano plaća — 80387 ima gotovo četiri puta više tranzistora nego 80386! Veći broj tranzistora označava veću potrošnju, pa će aritmetički koprocesor na vašem PS/2 80 disipirati mnogo veću snagu, što se odražava na njegovo zagrevanje — temperature keramičkog kućišta je tipično 44—55 stepeni Celzijusa. Ukoliko je koprocesor okružen čipovima koji dosta troše i ukoliko je ventilator slab, ova temperatura može da poraste i izađe izvan specifikacija, što daje izuzetno neprijatne rezultate — pri računanju se gubi po neki bit i to na nepredvidljiv način! Dalji porast temperature dovodi do kraha čitavog sistema, što znači da je kvalitet izrade najvažnija karakteristika koju aritmetički koprocesor treba da zadovolji — verovatno zbog toga samo Intel proizvodi ove čipove i verovatno su zato oni toliko skupi. Čak i renomiranoj firmi poput Intela proizvodnja aritmetičkih koprocesora donosi mnogo problema — procenat škarta je preko 80 posto, što znači da tek svaki peti čip prolazi stroge testove i izlazi na tržište!

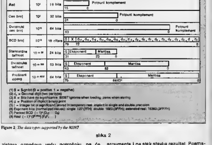


Figure 2 The data types supported by the 80387

slika 2

Pokušajmo da malo preciznije analiziramo arhitekturu koprocesora 80387 koja je prikazana na slici 1. 80-bitni registri opšte namene su, pre svega, organizovani na principu steka — u njih se upisuju podaci koji se sabiraju, oduzimaju, množe, dele i porede, odnosno argumenti elementarnih funkcija, a onda koprocesor sa steka skida argumente i na stek stavlja rezultat. Posmatrajmo, na primer, sabiranje: u registrima R0 i R1 će se nalaziti argumenti da bi se po izvršavanju naredbe FADD u R0 našao njihov zbir.

Brojevi se upisuju strogo u skladu sa standardom IEEE 754-1985 *(Standard for Binary Floating Point Arithmetic)*, raniji Intelovi aritmetički koprocesori su nastali pre standarda koji je propisalo renomirano internacionalno udruženje elektroinženjera (IEEE se izgovara kao *aj tripl i*) što je donelo oštre zamerke mnogih profesionalaca i korisnika većih kompjuterskih sistema. Zato je Intel ovoga puta, čuvajući maksimum vertikalne kompatibilnosti, usvojio standarde i uveo sedam tipova podataka koji se belažu u tri različita formata opisane na slici 2: binarni celi brojevi, binarno kodirani decimalni celi brojevi i binarno predstavljeni racionalni brojevi. Svi brojevi se interno prevode u dvostruku tačnost.

. . . i Formati zapisa

Što se samog formata zapisa tiče, svaki se broj sastoji od jednobitnog znaka (nula označava pozitivne, a jedinica negativne brojeve), 64-bitne mantise (novi Intelovi

334

priručnici, prilagođavajući se IEEE standardima postepeno zamenjuju reč mantisa sa *significand* ali se nama čini da bi u domaćim uslovima nepotrebna promena ustaljenog termina izazvala nepotrebnu zbrku) i 15-bitnog eksponenta. Vrednost 0 može da bude pozitivna ili negativna i sastoji se od proizvoljnog znaka i sedamdeset devet binarnih nula. Brojevi čiji se eksponent sastoji od samih jedinica su takođe specijalni: ako im je mantisa 1.00...00 radi se o „broju" beskonačno *(infinity)* koji, zavisno od znaka, može da bude „plus beskonačno" ili „minus beskonačno" — ovakva nomenklatura je sasvim u skladu sa IEEE standardima koji zahtevaju da, primera radi, 1/0 signalizira grešku ili da rezultat u odgovarajućem registru bude smislen i da ima vrednost „beskonačno". Ukoliko se eksponent sastoji od jedinica a mantisa je različita od 1.00...00 radi se o ne-broju (NaN ili *not a number*). 80387 poznaje „mirne" *(quiet)* i „alarmne" *(signaling)* ne-brojeve koji se razlikuju na osnovu najsignifikantnijeg bita mantise. Obzirom da su „mirni" ne-brojevi poznati i korisnicima 80287, ovoga puta ćemo posvetiti nešto prostora novitetu koji smo nazvali alarmni ne-brojevi.

Većina savremenih programskih jezika, po ugledu na paskal, ne dopušta programeru da koristi promenljive kojima prethodno nije dodeljena vrednost. Na prvi pogled bi se reklo da ovakav zahtev predstavlja nepotrebno ograničenje (na nekim verzijama bejzika referenciranje nedefinisane promenljive povlači njeno automatsko definisanje i dodeljivanje neke karakteristične vrednosti, obično nula), ali se u stvari radi o velikoj pogodnosti — zamislimo da smo promenljivu *suma* u nekom izrazu greškom nazvali *sume* — ako interpretator odnosno kompajler ne prijavi grešku, program će raditi pogrešno na način koji će nas možda dovesti do očajanja!

Kako se tehnički realizuje registrovanje nedefinisanih promenljivih? Kada se radi o interpretatoru, problema nema: u trenutku izvršavanja naredbe proverava se tekuća lista promenljivih i uvek se zna šta je definisano a šta nije. Kod kompajlera je, međutim, problem gotovo nerešiv: u toku prevođenja se statički alocira prostor za promenljive. Kompajler će jasno prijaviti promenljive koje su referencirane iako *nigde* nisu definisane, ali nema nikakvog načina da sazna da li će u toku izvršavanja programa neka promenljiva biti najpre pozvana a onda definisana! U pomoć priskaču alarmni ne-brojevi — kompajler svakoj promenljivoj dodeljuje istu ne-vrednost, a onda se aritmetički koprocesor „buni" čim program pokuša da izvrši bilo koju operaciju čiji argument nije ispravan. Rezultat — sigurnije programiranje i lakše pronalaženje grešaka!

Sigurnom programiranju pomaže i takozvani registar oznaka *(tag word)* samog koprocesora. Dobro nam je poznato da u računarstvu sadržaj nekog registra ne može biti nedefinisan — svaki bit jednostavno mora da ima vrednost nule ili jedan! Sadržaji nekih registara, sa druge strane, mogu biti logički nedefinisani — ako smo na stek stavili samo jedan broj, preostalih šest registara steka nema nikakve smislena vrednosti. Obično će se u takvim registrima naći nule, ali konstruktor svakog aritmetičkog koprocesora dobro zna da je nula broj

Test	80387	WTL 1167
Fbench	4.01	3.35
Float	1.03	0.35
Savage	2.84	2.63
Calcpi	3.11	0.99
Whetstone	0.57	0.35

slika 3

kao i svaki drugi; *nula* ne označava *ništa*. U nedefinisane registre bi, istini za volju, mogao da se upiše neki ne-broj, ali je prepoznavanje ne-broja daleko sporije od testiranja jednog flega — registar oznaka određuje tip podataka upisanog u svaki od opštih registara. Za kontrolu rada procesora zaduženi su i neizbežni statusni odnosno kontrolni registar.

Ostalo je još da pomenemo 48-bitne ukazatelje: ovi registri zapravo predstavljaju deo mikroprocesora 80386 i obezbeđuju prenošenje naredbi, podataka i rezultata između „mozga" i „računaljke".

Nove instrukcije

Najvažniji noviteti koje je 80387 ponudio na polju seta instrukcija su FSIN, FCOS i FSINCOS. Upotreba prve dve je očigledna: sa vrha steka se skida argument u radijanima (mora biti manji od 2^63) a onda se na vrh steka postavlje vrednost njegovog sinusa odnosno kosinusa. Treća naredba je jedva nešto složenije: na stek se smeštaju vrednosti sinusa i kosinusa zadatog argumenta, naredba predstavlja dobrodošlo ubrzanje pri radu sa grafikom, gde je često neophodno izračunati sinus i kosinus istog ugla.

Ostatak novih instrukcija svodi se na razna poređenja. FUCOM je slična postojećoj instrukciji FCOM (poređenje racionalnih brojeva), a tim što se u slučaju ne-broja ne prijavljuje greška. FUCOMP i FUCOMPP takođe imitiraju FCOM ali skidaju jedan odnosno dva operanda sa steka, dok FPREM1 obezbeđuje računanje ostatka pri deljenju u skladu sa IEEE standardima — računa se ST/ST(1) pri čemu se količnik smešta u ST a ostatak u ST(1).

Osim uvođenja novih instrukcija, 80387 je umanjio ograničenja postojećih naredbi koprocesora 80287: FPTAN, FPATAN, F2XM1 i FSCALE — FPTAN *(partial tangent)* je, na primer, primao argumente između –PI/4 i +PI/4 a sada se snalazi i sa brojevima čija je apsolutna vrednost manja od 2^63. Uvedene su, zatim, konstante 0, 1, PI, LOG 10, LOG e, LOG 2 i LN 2 koje se dovode na stek komandama FLDZ, FLD1, FLDPI, FLDL2T, FLDL2E, FLDLG2 i FLDLN2 — tačnost svake konstante je približno 19 cifara odnosno 64 bita.

Novina je, najzad, i detaljnije prijavljivanje greški — uvedene su kodirane poruke *Invalid operation* (stek je prekoračen, neki od operanada je alarmni ne-broj ili izlazi iz dopuštenog opsega), *Devide by zero* (deljenje nulom koje se može maskirati kao i većina drugih greški), *Denormal operand* (neki od argumenata predstavlja ne-broj ili broj bez smisla), *Numeric overflow or underflow* (prekoračenje opsega) i *Inexact result* (operacijama sa brojevima bitno različitog reda veličine izgubljeno je previše značajnih cifara).

Na brzinskom testu

Postoje samo dva načina da upoznate brzinu aritmetičkog koprocesora — teži (i sigurniji) zahteva kupovinu logičkog analizatora i brojanje ciklusa, dok se lakši (i nesigurniji) svodi na pisanje brzinskih testova, takozvanih benčmark *(benchmark)* programa.

Pisanje programa koji testiraju brzinu nekog računara je vrlo osetljiva operacija — ukoliko znamo kako neki sistem funkcioniše, lako ćemo napisati program koji će na njemu biti mnogo brži nego na drugim mašinama, kao i program koji će na njemu biti mnogo sporiji — zato mnoge firme biraju brzinske testove koji odgovaraju baš njenim proizvodima! Čak i ako je brzinske testove pisao nepristrasni programer, u njihove rezultate ne smete previše da se pouzdate — možda su isprobane baš stvari koje su vama retko potrebne. Sve brzinske testove ćemo, dakle, uzeti sa velikim rezervama, ali bez njih ipak ne možemo — bolje je, na kraju krajeva, imati bilo kakve nego nikakve podatke!

Slika 3 prikazuje rezultate pet standardnih benčmark testova časopisa *Byte* koji testiraju performanse aritmetičkih koprocesora. Programi su izvršavani na Compay-ovom računaru DeskPro 386 koji radi na 20 MHz. Testovi *Float* i *Calcpi* (računanje broja PI preko Tajlorovog reda) se intenzivno bave operacijama sa racionalnim brojevima. *Whetstone* i *Fbench* predstavljaju kombinaciju rada sa racionalnim brojevima i računanja vrednosti transcedentnih funkcija, dok *Savage* računa isključivo vrednosti transcedentnih funkcija. Primera radi, u istoj su tabeli dati i rezultati koje je postigao WTL 1167, aritmetički koprocesor firme Weitek koji se takođe povezuje sa modelom 80 iz serije PS/2 odnosno računarom DeskPro 386; treba znati da je WLT 1167 itekako profesionalan uređaj koji se sastoji od čitave (ko)procesorske ploče i košta preko 2000 dolara — četiri puta jeftiniji 80387 se nije pokazao za red veličine sporijim! Ukoliko ove rezultate uporedimo sa podacima iz „Računara 29" i „Računara 35", videćemo koliko je 80387 brži od 8087 i 80287. Za detaljnije poređenje trebalo bi, jasno, uzeti u obzir i rezultate koje postižu parovi procesor/koprocesor drugih proizvođača (mislimo prvenstveno na Motorolu), ali ove podatke ostavljamo za drugu priliku — radoznali mogu da pročitaju tekst „Sprinteri na duge staze" iz „Računara 32".

Treća generacija Intelovih aritmetičkih koprocesora sve u svemu predstavlja gotovo dramatičan korak napred u odnosu na drugu koja je sa svoje strane bila prilično slična prvoj. Verujemo da će aritmetički koprocesor 80387 približiti IBM-ove računare iz serije PS/2 i njihove klonove mnogim primenama koje su do skora bile rezervisane samo za mnogo skuplje i moćnije grafičke radne stanice. Šteta je jedino što veoma visoka cena koja neće uskoro pasti uklanja 80387 iz domena „smrtnih" korisnika.

Dejan Ristanović

Programi u javnom vlasništvu

Javni program

Izdavači komercijalnog softvera ne biraju sredstva da spreče neovlašćenu distribuciju svojih programa. Postoje, međutim, i organizacije kojima je cilj da besplatno šire svoje programe. Radi se o tzv. programima u javnom vlasništvu koji od nedavno mogu da se, na organizovan način, nađu i u našoj zemlji.

Public domain (javno vlasništvo) je jedna od dvije glavne vrste programa koje zajednički zovemo freeware (kovanica od engl. free=besplatan i ware=roba). Druga vrsta su tzv. user-supported, odnosno shareware programi.

Kao što i samo ime freeware kaže, radi se o programima koji su besplatni.

Koncept freeware-a (žao mi je, ali nikako ne mogu naći odgovarajući prijevod za ovo) je razvio Andrew Fluegelman. U konceptu freeware-a leži ideja da kopiranje programa treba ohrabrivati (ne, ne, dragi moji pirati, nije to ono što ste vi mislili, prerano ste se obradovali) umjesto ograničavati, odnosno da vrijednost nekog programa najbolje može ocijeniti njegov korisnik, te da bi računarski mije trebao imati interesa u podržavanju razvoja korisnog softvera.



U interesu korisnika

Kako Programi u javnom vlasništvu imaju dosta zajedničkih ali i dosta različitih (naročito što se tiče prava korisnika) tačaka sa programima koje podržavaju korisnici (usersupported).

Kod programa u javnom vlasništvu autor programa polaže mala ili nikakva prava na programe koje je napisao. Programi se mogu koristiti ili mijenjati sa malim ili nikakvim ograničenjima. Program možete kopirati i davati drugima u neograničenim količinama, dokle god to nije na komercijalnoj osnovi. Od vas se pri tom očekuje jedino *fer plej*. Kako ovi programi nisu pisani ni na kakvoj komercijalnoj osnovi, niti se od njih očekuje da donesu neki prihod svom tvorcu, ne očekujte nove verzije, niti bilo kakvu drugu podršku.

Programi koje podržavaju korisnici mogu se također kopirati prijateljima, a ovi ga besplatno koristiti koliko žele, ali autor programa polaže sva legalna prava na njega. Dakle, nije dozvoljeno komercijalno prodavanje programa niti njegovo mijenjanje.

Ukoliko vam se program sviđa, ukoliko ga smatrate korisnim, autora možete donirati nekim iznosom (obično je u programu dana adresa autora i iznos za koji autor smatra da bi bio zadovoljavajući, a uglavnom je to od 5 do 50 dolara) ili se za određenu sumu (koja je dosta veća od predložene donacije, ali još uvijek i značajno manja od cijene komercijalnih programa iste vrste) možete registrirati kao korisnik.

Registrirani korisnici dobijaju razne beneficije kao što su:

— mogu od autora dobiti kompletnu dokumentaciju (dokumentacija se inače nalazi na disketi u obliku fajla spremnog za štampanje), ovo ponekad uključuje i izvorni kod;

— mogućnost da se telefonski obratite autoru za pomoć;

— da dobijete nove verzije programa i obavještenja o njima.

Registrirani korisnici ponekad mogu dobiti i pravo da program prodaju na osnovu licence za određen broj kopija ili određenu svotu novca za svakog korisnika koji se registrirao nakon što je program kopirao od njih.

Za programe koje podržavaju korisnici biblioteka programa u javnom vlasništvu, kao i besplatno kopiranje, služe samo kao način distribucije programa bez troškova na marketing.

Ovi programi žive onoliko dugo koliko vlada interes za njih. Dakle, dokle god korisnici doniraju autora i registriraju se, autor će naći razlog da program održava, unapređuje i pomaže korisnicima. Korisnici će opet biti zadovoljni jer će za male pare doći do kvalitetnog programa koji im je potreban. Odatle ovim programima i ime: programi koje podržavaju korisnici.

Za one koji sumnjaju u kvalitet ovakvog softvera, moram reći da su mnogi programi u ovoj grupi profesionalne kvalitete. Neki od njih su zajedno sa svojim programerima već odavno ušli među legende računarstva (o njima ćemo kasnije reći još nekoliko rečenica).

Pogled unazad

Sada, kada smo raščistili sve pojmove, da zavirimo malo u povijest ove pojave.

U početku razvoja računarstva (pri tome ne mislim na računarstvo uopće, nego na onu njegovu granu gdje se koriste tzv. lični računari) nije postojao komercijalan softver, niti njegovo tržište. To nije ni čudno kao se zna da je većina prvih ličnih računara nastala pod rukama i lemilicama njihovih vlasnika. Lični računari su tada bili jednako egzotični i rijetki kao što je bio i automobil u svojim prvim danima.

Sretni vlasnici, tj. programeri, bili su prisiljeni pisati vlastite programe za svaku stvar koja im je zatrebala. Tako se javljala sve veća potreba za programima kojih nije bilo (taj osjećaj poznajete ukoliko posjedujete neki egzotičan ili zastarjeo računar). Nicanjem klubova stvorila se prilika za međusobno upoznavanje i druženja vlasnika računara, čime su stvoreni preduvjeti za razmjenu programa, do čega je uskoro normalno i došlo. Na taj način je rođen softver u javnom vlasništvu.

Dva su događaja utjecala na to da se razvoj stvari počne odvijati eksponencijalnom brzinom. Prva je pojava CP/M-a kao standarda. Začarani krug (više računara koji prihvataju iste programe=> više programera koje rade programe za te računare=> više programa=> više ljudi koji kupuju isti računar radi tih programa=> više računara koji prihvataju iste programe) je tako, na radost svih, zatvoren.

Drugi podstrek razvoju je bila pojava XMODEM protokola (često ga zovu i Christensenov protokol, po njegovom tvorcu Wardu Christensenu) za telekomunikacije. To je omogućilo prenošenje programa na daljinu i razvoj *billetin board* sistema. Time vlasnici određenog računara nisu bili u razmjeni ograničeni samo na svoju neposrednu fizičku okolinu, pa je ubrzana cirkulacija i širenje programa, a *billetin board* sistemi su postali prava rasadišta programa u javnom vlasništvu.

Crvena konjica

Kako uz svaku historiju idu i legende, tako je i sa ovom historijom. Jedina razlika je što su ovdje sve legende istinite. Dakle, da vidimo tko je sve i kako postao besmrtan.

Prvi program koji je bio razvijen i prodan pod freeware konceptom je Fluegelmanov PC-TALK. Radi se o jednom od najboljih programa za komunikacije koji se nalaze u javnom vlasništvu. Priča se da je pukovnik North (sjetite se nedavne američke afere sa Iranom) svoje komunikacije sa pretpostavljenima obavljao upravo sa ovim programom.

Najpoznatiji shareware tekst procesor je svakako PC-WRITE. Autor mu je Bob Wallace (nekadašnji programer u Microsoftu). Uz pomoć redovnih oglasa u „BYTE“-u i dosta pohvalnih riječi koje je o autoru i programu izrekao „Byte“-ov poznati kolumnista Jerry Pournella (u njegovoj stalnoj rubrici „Computing at Chaos Manor“), ovaj program je okružen svojevrsnom famom koja njegovom autoru donosi sasvim pristojne prihode i omogućava normalan život bez obzira što nije zaposlen.

Isti status uživa i CORE WAR. Ova igra (nisam siguran da je ovo adekvatan naziv, ali ostavimo

tu raspravu za kasnije) je prilično popularne na Zapadu a i u nas polako počinje da stiče svoje pristalice. Njena implementacija za PC je opisana u časopisu SCIENTIFIC AMERICAN (članak „Computer Recreations", maj 1984) a napisao ju je Kevin Bjorke koristeći SMALL-C.

RED RYDER je najbolji program za komunikaciju na „mekintošu". Postigao je veliku popularnost što mu je uz kvalitet omogućilo da postane i isplativ svojim tvoricima. Na žalost jedan od njih nije to doživeo. Uz ovaj program su vezane još i dvije, nazovimo ih, polulegende. Jedna se tiče njegova imena (morate priznati da je „Crveni jahač" poprilično čudno ime za jedan komunikacioni program) i kako je ono nastalo, a druga je nastala kao rezultat nastojanja njegovog preživjelog tvorca da ljude odvrati od takvih stvari kao što su provale u kompjuterske sisteme. Zato je u program ugradio nekoliko zagonetki i objavio da će nagraditi prvu osobu koja uspije da ih riješi. Usput je i natuknuo da zagonetki ima 10. Prava istina leži u tome da ih je samo tri i zato se može shvatiti odakle mu onaj tajanstveni osmijeh na licu kada mu s vremena na vrijeme pristupi neki mladi „haker" i izjavi nešto poput toga da je već riješio 9 zagonetki i da mu ostaje još samo jedna.

SMALL-C je svakako najpoznatiji program koji se nalazi u javnom vlasništvu. Radi se o kompajleru za c koji je podskup standardnog C-a kakvog su opisali Kernighan i Ritchie. Autor mu je Ron Cain, a pisan je u C-u. Od prvog dana njegovog objavljivanja u „DR. DOBB'S JOURNAL"-u dobio je brojne pristalice pa postoje brojne verzije za razne računare kao i nekoliko biblioteka pomoćnih funkcija. Na disketama se uz njega često dobija i izvorni kod.

Biblioteke u svijetu

Izbor mjesta na kojima se može nabaviti softver u javnom vlasništvu je dosta velik. Ako se izuzmu biblioteka programa u javnom vlasništvu, onda među svima prednjače klubovi. Grubo podijeljena postoje dvije vrste klubova: (a) klubovi korisnika određenog tipa, pa čak i modela računara i (b) klubovi korisnika sa specijalnim interesima. O prvima ne treba pričati jer se o njima može sve zaključiti i iz samog imena. Ovi drugi okupljaju korisnike sa specijalnim interesima, pa se često uz njihovo ime može naći i skraćenica SIG (Special Interest Group). U njima se mogu naći ljudi koje zanima tačno određena tema u računarstvu (npr. C-jezik, umjetna inteligencija, TURBO PASCAL itd.). Ove grupe su raj za svakog programera jer svi programi koje one nude dolaze u obliku izvornog koda.

Kao što se vidi, izbor je doste bogat, ali ja još nisam uspeo locirati niti jednu specijalnu grupu sa interesom za MACRO ASSEMBLER na PC računarima, pa ovim putem molim čitaoce da mi pomognu u potrazi i svoje rezultate jave redakciji.

Najpoznatiji i najstariji klub za PC računare je PC-SIG. Osnovan je 1982. Na svom spisku, prema podacima iz marta 1987, drži preko 700 disketa sa programima za PC. Zastupljeni su programi iz sljedećih grupa:

- — umjetna inteligencija
- — software za vođenje bulletin board sistema
- — poslovne aplikacije
- — komunikacije
- — baze podataka
- — obrazovanje uz pomoć kompjutera
- — programi za organiziranje radne okoline,
- — programi za zabavu i igre
- — financijske aplikacije
- — grafika
- — kućne aplikacije,
- — kompjuterski jezici
- — matematički programi
- — inženjerski programi
- — specijalne aplikacije
- — unakrsne tabele
- — programi za obradu teksta,
- — programi za backup
- — programi za šifriranja i dešifriranja podataka
- — i razni utility programi

337

Željeni programi se mogu dobiti na disketama, ali i na CD-ROM disku. Normalno, u tom slučaju ne dobijate samo jedan program, nego cijelu biblioteku!

Procjenjuje se da PC-SIG uspijeva, uz česte oglase u vodećim računarskim časopisima, plasirati programe do preko pola miliona korisnika!

Za sve detaljnije informacije se možete obratiti na adresu: PC-SIG INC., 1030 E. DUANE AVENUE, SUITE D, SUNNYVALE, CA 94086 USA.

Zadnji u spisku mjesta na kojima se može doći do željenih programa su *bulletin board* sistemi (jedna vrsta elektronskih oglasnih ploča) i sistemi za telekomunikacijske konferense. Na ovim mjestima se nalazi, izgleda, nepresušno vrelo neprocenjivih informacija, jer sve firme koje u računarstvu nešto znače i koje drže do sebe, imaju *bulletin board* sisteme do kojih imaju pristupa svi bez ograničenja. Među njima su, da navedemo samo neke od najpoznatijih, „BYTE", „DR. DOBB'S JOURNAL", „Steve Ciarcia" („BYTE"-ov čarobnjak za hardver) itd.

Bulletin board sisteme ne drže samo firme nego i pojedinci — entuzijasti, jer je cjelokupna oprema po cijeni dostupna prosječnom Amerikancu. Međutim, na privatnim *bulletin board* sistemima se čovjek može lako i opeći, jer su česta mjesta iživljavanja „programera" sa čudnim smislom za „humor".

Kako su kontakti sa ovim sistemima praktično (zbog cijene telefonskog razgovora) nedostupni prosječnom Jugoslavenu, ne mislimo da je potrebno da dajemo njihovu adresu, odnosno broj telefona. Ukoliko se ipak pokaže da postoji interesovanje za njih kod većeg broja čitalaca, željene brojeve ćemo objaviti.

... i kod nas

U nas softver u javnom vlasništvu sve donedavno praktično nije postojao. Jedan od razloga zašto je to tako je i to što su svi programi bilo kakve vrijednosti koji bi jednom dospjeli do naše zemlje praktično istog trenutka prelazili u jednu drugu vrstu „javnog vlasništva". O ovoj temi nećemo ovdje raspravljati, što ne znači da o njoj ne bi trebalo govoriti, jer nam ova pojava sve više ruši ugled u svijetu i nanosi nam veliku štetu.

Bez obzira na sve to, dva primjera koja ćemo navesti pokazuju da postoje ljudi koji misle i rade drukčije. Zbog svog entuzijazma zavrijedili su da ih se ovdje predstavi, a naši čitaoci sada neće više trebati trošiti tako dragocjene devize.

Dva su mjesta na kojem se može nabaviti softver u javnom vlasništvu u Jugoslaviji. Prvi je Ljubljana, a drugi je Zagreb.

U Ljubljani je to ADIN KRUG oko kojeg se okuplja skupina volontera. Krug je u augustu 1987. osnovala radna organizacija MIKRO ADA. MIKRO ADA se bavi proizvodnjom računarske opreme, savjetovanjem i obučavanjem. Princip rada je totalno amaterski, tj. nije komercijalan, a novac dobijen od članarina u klubu i prodaje disketa sa programima pokriva troškove. Nakon pristupanja Krugu (polugodišnja članarina je za radne organizacije 36000.— dinara, a za privatna lica 12000.— dinara), članovi redovno (svaki mjesec) dobijaju kataloge s informacijama o novim programima, te imaju prednost i popust od 20% pri nabavi programa. Cijena jedne diskete s programima je u januaru iznosila 7000.— dinara. Kada se od toga odbije popust, ispada da član Kruga dobija disketu punu programa za 5600.— dinara (po cijeni iz januara). Svima onima kojima se učini da je to mnogo, preporučujem da pogledaju cijene disketa u trgovini. Kad vide da prazna disketa u trgovini koštaju koliko i puna u ADINOM KRUGU, uvidjeće da ovaj softver dobijaju praktično besplatno. U cijenu normalno nije uračunata poštarina, a rok slanja je 5 dana. Diskete možete podići i sami u prostorijama MIKRO ADE.

Programi kojima raspolaže ADIN KRUG su uglavnom iz inozemnih izvora (PC-SIG, BIX, COMPULINK), ali ima i vrlo kvalitetnih programa koji su djelo momaka iz Kruga. Klub zasada raspolaže sa oko 30 disketa sa programima, što nije mali broj kada se zna da su svi ovi programi marljivo probrani i testirani, te prevedeni za naše govorno područje. Koliko mnogo znači kad je program preveden, to svatko zna, ali malo tko, ako nije poslovao sa PC-SIG-m i ostalima, i ako nije vidio spisak svih programa PC-SIG-a od preko 400 strana pisan sitnim slovima, može znati šta to znači kad je netko program testirao i ocijenio umjesto tebe. Kada ne bi bilo ovog probiranja, uz konstantni mjesečni priliv od oko 100 programa u Krug, kvantitet bi progutao kvalitet, a to nikome nije želja.

Nakon svega ovoga, možda ćete se pitati: učlaniti se ili ne? Što se tiče mene — ja sam se već učlanio! Zašto? Zato jer smatram da za kvalitet treba i platiti, a momci iz Kruga nude vrlo dobar kvalitet uz cijenu koja je svakome prihvatljiva.

Adresa ADIN KRUGA je: MIKRO ADA, ZA ADIN KRUG, CANKARJEVA 10B, 61000 LJUBLJANA, TEL. (061) 219-125.

Usput, kao ilustraciju ozbiljnosti kojom je pristupljeno ovom poslu, navešću da sam prospekt s informacijama o načinu učlanjenja u Krug dobio 48 sati nakon što sam ga zatražio, što znači da je poslan istog trenutka nakon što je moja narudžba primljena! Drugi izvor javnih programa u Jugoslaviji je DAM DATA, servis za elektronsku obradu podataka, servisiranje računara i izradu oprema i programa za njih.

U DAM DATA servisu, osim navedenih usluga, nude i javni softver. Programi potiču iz tri izvora: PC-SIG-a, PC-BLUE biblioteke i biblioteke korisnika C-a. Programi u prvoj grupi potiču sa disketa PC-SIG sa rednim brojevima od 1 do 568, u drugoj sa disketa sa rednim brojem od 1 do 200, a u trećoj sa rednim brojem od 8101 do 8240. Sve u svemu, na raspolaganju je oko 1000 disketa, što je impresivan broj! Najviše poštovanja u meni izaziva bogat izbor disketa iz treće grupe, jer sadrže izvorni kod svih danih programa u C-u. Iako nisam ljubitelj C-a (suviše je kriptičan za moj ukus), moram priznati da diskete iz ove grupe imaju neprocjenjivu vrijednost za svakog programera.

Što se tiče cijena, one su 14000.— 13000.— i 15000.— dinara po disketi za diskete iz prve, druge i treće grupe. Navedene cijene su za radna organizacija, pa je u slučaju plaćanja gotovinom za više od 10 disketa cijena 9000.— dinara po disketi. Za druge eventualne popuste trebate se raspitati direktno. U cijenu je uračunata diskete, troškova pakiranja i poštarina.

Adresa na koju možete dobiti opširnije informacije je: DAM DATA, MARINKO KRIŠTO, FRANZA MEHRINGA 5, 41000 ZAGREB.

Na kraju balade

Ova pojava nije kratkog daha. Na Zapadu je već odavno dobila status punopravnog građanina računarskog svijeta i ni najeminentniji časopisi (kao što je, na primjer, „BYTE") ne stide se da na svojim stranicama opisuju i spominju freeware programe. Štaviše, odnose se prema njima s poštovanjem. Osim toga, nema toga tko ovim programima može zanijekati kvalitet ili omalovažiti stručnost i rad uložen u njih, jer samo dobar stručnjak može napisati nešto kao što je SMALL-C, XLISP, PD-PROLOG i slično.

Autor teksta je veoma optimističan što se tiče sudbine freeware-a u Jugoslaviji. Smatramo da je on dobra alternativa piratstvu (u inostranstvu se ova pretpostavka već odavno pokazala kao tačna), jer je za svakog Pravog Programera veliki izazov napisati program kojem će se diviti njegove kolege i u zemlji i u inostranstvu. Osim te moralne satisfakcije i prilike da bude poznat, priznat i koristan u računarskom miljeu, pružena mu je i prilika da zaradi i koji dinar, odnosno dolar, jer ovakav koncept omogućava autorima programa njihovu distribuciju bez ikakvih troškova.

Inostranstvom već kruži nekoliko freeware programa koji su djelo jugoslovanskih autora, a kada ovo budete čitali, u američkoj centrali PC-SIG-a će ih biti još. Pridružite nam se!

Zoran Cvijetić, dipl. ing.

DOS 3.30 na prvi pogled

Naslov ovoga teksta može da vas navede na pomisao da su vam utrapili pola godina stare „Računare" — PC DOS 3.30 je promovisan pre skoro godinu dana i stigao u Jugoslaviju jedva nešto posle toga; prvi pogled se, kako izgleda, otegao! Činjenica je, medutim, da je do skora bilo potpuno nemoguće doći do opsežnije dokumentacije za DOS 3.30 što znači da su mnogi noviteti koje on donosi ostali skriveni.

Svaki vlasnik IBM PC-ja zna koliko je teško promeniti DOS Problem nije samo u formatiranju hard diska — neki programi koji su se savršeno snalazili sa starim DOS-om možda zahtevaju ponovno instaliranje, neka od divljih rutina koje smo pisali ne funkcionišu nikada se ne zna koji će se bagovi kada ispoljiti Dugom životu neke verzije DOS-a svakako pogoduje inercija ali i saznanje da nova verzija u suštini i nije toliko različita od stare — DOS 2 11 i 3 30 se formalno mnogo razlikuju ali će opsežnija analiza pokazati da je brojna razlika isključivo kozmetičke prirode Stoji najzad, i argument zašto da predem na DOS 3 10 kad će se uskoro pojaviti DOS 3 20

Skromni noviteti

Jugoslavenski PC-ji sve u svemu predstavljaju živu istoriju MS DOS-a — okrenite se unaokolo i naći ćete desetak verzija! Primat ipak ima DOS 3 10 — bolji poznavaoci PC-ja čak tvrde da je on bolji i pouzdaniji od DOS-a 3 20 Obzirom da je DOS 3 30 po prilici poslednja verzija MS DOS-a (IBM i Microsoft polako prelaze na OS/2) prelazak na njega je odluka koju treba pažljivo razmotriti

Na početku ćemo pobrojati novitete koji su vam verovatno poznati — IBM ih je veličao u propagandnim materijalima koji su pratili lansiranje računara iz serije PS/2 DOS 3 30 pre svega, podržava i hard diskove veće od 32 megabajta ali je ta podrška krajnje ograničena — disk od 40 megabajta na primer, možete da podelite na dva logička diska od po 20 megabajta ili jedan disk od 32 i jedan od osam megabajta Ovu smo opciju isprobali na AT klonu koji je opremljen internim hard diskom od 20 megabajta i eksternim diskom od 40 M Podela funkcioniše savršeno ali nije potpuno kompatibilna sa prilično poznatim *public domain* programom Hard Split koji smo do sada koristili kako bismo podelili 40 M na dva puta po 20 Red sa DOS-om 3 30 je nešto pogodniji jer ne zauzima dodatni RAM poput rezidentnog Hard Split-a Hard Split je bolji jer je kopiranje sa diska D na disk E bitno brže i jer bolje saraduje sa našim omiljenim programom Norton Utilities (prvi utisci o kooperaciji DOS-a 3 30 i Nortona mogli bi da zauzmu par strana u „Računarima" ali ćemo ih izložiti nekom drugom prilikom — moguće je da našoj (piratskoj) verziji Nortona nešto nedostaje) Pokazuje se najzad da DOS potpuno ne „razume" particiju koju je Hard Split napravio — tek kada smo disk formatirali sve je došlo na svoje mesto



Ostali opšte poznati noviteti svode se na BAT datoteke koje mogu da pozivaju jedna drugu (praktični značaj ovog sa teorijske strane velikog unapredenja je krajnje ograničen — jezik koji se koristi pri pisanju BAT programa je suviše rudimentaran da bi na njemu mogli da se razvijaju veći programski paketi bazirani na modularnosti) i časovnik realnog vremena koji može da se doteruje se *date* i *time* (naš AT je ovo dopuštao i dok smo radili sa DOS-om 3 10)

Disk

Novina koja nam se neobično svidela je naredba FORMAT Svi znamo da je PC u svojoj karijeri promenio nekoliko formata disketa i da je naredba FORMAT stalno dobijala nove i sve manje razumljive opcije — autor ovoga teksta se na primer nikada nije bio siguran zbog čega se formatiranje disketa od 360 K na drajvu od 1 2 megabajta zahteva sa FORMAT **A /4. DOS** 3 30 konačno dopušta da otkucate FORMAT A: /N xx /T=**yy** gde **yy** predstavlja broj treka (npr 40 ili 80) a **xx** broj sektora na traci (npr 8 ili 9) Pravo je čudo što Microsoftovi programeri nisu predvideli parametar /S koji bi opisao da li je disketa jednostrana ili dvostrana što znači da će oni koji iz nekog razloga žele arhaične diskete od 160 K morati da koriste staru opciju /1 Još je veće čudo da sve ove godine niko nije napisao pristojan program FORMAT koji bi na ekranu ispisao meni, detaljno objasnio razne formate i omogućio da izaberemo onaj koji nam je potreban

DOS 3 30 u potpunosti podržava disk jedinice od 3 5 inča pa čak i razne njihove kombinacije U tom smislu je posebno značajna deklaracija DEVICE = DRIVER SYS /D ddd /T ttt /S ss /H hh /C /N /F f koja se ugraduje **u** datoteku CONFIG SYS **ddd** je oznaka drajva (broj a ne slovo što je prilično zbunjujuće, prvi fizički disk drajv nosi oznaku 0 dok je oznaka prvog hard diska 128) **ttt** je broj traka **ss** broj sektora **hh** broj glava (dvostrana flopi disk jedinica ima dve glave) /C obezbeduje kopiranje sa diskete na disketu pri čemu se koristi jedan drajv a računar s vremena na vreme zahteva zamenu disketa /N označava nepomičnu jedinicu spoljne memorije kad što je hard disk a f predstavlja neku vrstu džokera koji po potrebi postavlja podrazumevane *(default)* vrednosti ostalih parametara Podrazumevane vrednost za f je 0 što opisuje standardan drajv od 360 K Zanimljivo je da je Microsoft uveo sličnu komandu (zvala se DRIVPARM) u DOS 3 20 ali ju je, kako izgleda odmah preimenovao

i video drajveri

Pomenućemo i datoteke DISPLAY SYS i KEYBOARD SYS Prva obezbeduje podršku EGA VGA i LCD ekrana — konačno će taster *Print Screen* preneti grafiku na papir čak i bez posebnih rezidentnih programa Sintaksa je DEVICE = DISPLAY SYS CON = tip hwcp n **Tip može da bude** MONO CGA EGA ili **LCD, hwcp** je (neobavezna) memorijska stranica (dozvoljene vrednosti su 437 850 860 863 i 865) a **n** odgovara broju specijalnih kodovi koji su definisani Nesumnjiva mana ove deklaracije je što ne podržava Herkulse standard koji IBM i dalje ne priznaje

KEYBOARD SYS je datoteka sa raznim definicijama tastature umesto desetina KEYBxx drajvera DOS 3 30 ima samo jednu komandu KEYB čiji je parametar oznaka države (Jugoslavija naravno, nema) i koja ostale parametre uzima iz konfiguracione datoteke

Pomenimo na kraju jedan bag DOS-a 3 20 koji je na novijoj verziji (skoro) otklonjen Naredba MODE LPT1=COM1 treba da sav tekst koji se šalje štampaču preusmeri na serijski port ali DOS 3 20 na našem AT-u jednostavno ignoriše ovu zapovest (DOS 3 10 je radio savršeno!) Bili smo vrlo zadovoljni kada je DOS 3 30 korektno preusmerio izlaz ali se naše zadovoljstvo pomalo istopilo dok smo istu komandu isprobavali na jednom originalnom (doduše industrijskom) IBM-ovom AT-u — MODE LPT1=COM1 je radilo ali MODE LPT1=COM2 i MODE LPT2=COM1 nije Očigledno je da „ima nešto trulo" u PC DOS komandi MODE ali bi bilo potrebno mnogo više istraživanja da se zaključi da li je problem u DOS-u BIOS-u, nekoj od kartica ili nečem desetom

PC DOS 3 30 sve u svemu, predstavlja sasvim umereno poboljšane prethodne verzije DOS-a koja svakako treba preporučiti novopečenim vlasnicima PC-ja Starim PC-jevcima njegovo instaliranje neće škoditi ali će se docnije možda pokazati da su primetna poboljšanja sasvim malobrojna

Dejan Ristanović

ORACLE®

RELACIONI SISTEM ZA UPRAVLJANJE BAZAMA PODATAKA I PORODICA SQL PROGRAMSKIH ALATA

U Računalniškom inženiringu KOPA se veruje da će u sledećih pet godina uspeh upravljanja organizacijama zavisiti pre svega od novih tehnologija, mikroelektronike, baza podataka i povezivanja računara. Zato se KOPA pobrinula da programski proizvodi ORACLE budu već danas na raspolaganju i našim, jugoslovenskim organizacijama.

Relacionim sistemom za upravljanje bazama podataka ORACLE i njegovom porodicom integrisanih SQL programskih alata završava se period robovske zavisnosti od određene marke računarske opreme. Programi napravljeni sa ORACLE jednostavno su prenosivi sa personalnog računara na mnoga druge mikro, mini i velika računare. Ujedno ORACLE povezuje i računare različitih proizvođača. **ORACLE radi na svim istaknutijim računarima, radnim stanicama i XT/AT kompatibilnim računarima, domaćih i stranih proizvođača** (ISKRA DELTA, EI-HONEYWELL, KOPA, IBM, DIGITAL, BULL, SIEMENS, DATA GENERAL, PRIME, NIXDORF, NORSK DATA, OLIVETTI, HEWLETT PACKARD, UNISYS, STRATUS, NCR, SEQUENT, WANG, APOLLO i SUN itd.)

Najveća prednost ORACLEA je brzo učenje i jednostavno korišćenje. Podaci su, naime, predstavljeni u obliku tabela, što prvo pojednostavljuje projektovanje baza podataka. A pri utvrđivanju potreba za informacijama olakšava komuniciranje između stručnjaka AOP i korisnika podataka i informacija.

ORACLE RDBMS je relacioni sistem za upravljanje bazama podataka. Dopunjava ga porodica integrisanih programskih alata SQL. Pojedinačni elementi mogu se skoro proizvoljno sastavljati i dopunjavati.

Prva verzija ORACLEA je bila instalirana već 1979. godine, a danas su proizvodi ORACLE vodeća tehnologija među relacionim sistemima za upravljanje bazama podataka na svetu. Stručnjaci Računalniškog inženiringa KOPA zajedno sa ORACLEOM EUROPE uvode, pružaju tehničku pomoć i održavanja proizvoda ORACLE v Jugoslaviji. **Ponosni su što domaćim korisnicima mogu da ponude programske proizvode takvih svojstava kao što ih ima ORACLE.** To su:

- **prenosivost programa nezavisna od vrste aparaturne opreme**
- **prototipski način rada**
- **potpuna kompatibilnost sa IBM-ovim SQL/DS i DB2**
- **mogućnost povezivanja i stvaranja distribuirane obrade podataka**
- **omogućava standardizaciju programske opreme**
- **omogućava veću produktivnost programiranja**

SQL * PLUS je jezik četvrte generacije sa kompletnom implementacijom IBM-ovoga standardnog jezika SQL.

SQL * FORMS je alat četvrte generacije koji omogučava brz razvoj programa koncipiranih na maskama.

SQL * REPORT je generator ispisa, koji omogućava brzu izradu različitih izveštaja.

SQL * MENU omogučava izradu menija za jednostavno povezivanja korisnika sa programima ORACLE i drugim programima.

SQL * NET omogučava komunikacija među procesima ORACLE na različitim računarima. SQL * NET omogučava zaista distribuiranu obradu podataka.

SQL * CONNECT omogućava povezivanje ORACLE sa podacima u bazi na drugim računarima koji koriste DB2 i SQL/DS.

EASY * SQL omogućava korištenje SQL početnicima i povremenim korisnicima uz pomoć jednostavnih menija.

SQL * GRAPH je alat koji omogučava kolor prikazivanja podataka u obliku raznih dijagrama.

SQL * CALC omogućava jednostavan pristup podacima u bazi.

PRO COBOL, PRO C, PRO FORTRAN, PRO ADA, PRO PL/I i PRO PASCAL su programski interfejsi između ORACLE i navedenih programskih jezika.

Pridružite se korisnicima ORACLEA, kojih ima više od šest hiljada u svetu. Među njima su i CIBA-GEIGY, HOECHST, DU PONT, BMW, FORD, GENERAL MOTORS, JAGUAR, RENAULT, VOLVO, DAIMLER BENZ, BOEING, MCDONNELL-DOUGLAS, NASA, AT & T, BRITISH TELECOM, ITT, SWISS BANK, CREDIT LYONNAIS i drugi.

ORACLE je zaštitni znak Oracle Corporation. ISKRA DELTA, EI-HONEYWELL, KOPA, IBM, DIGITAL, BULL, SIEMENS, DATA GENERAL, PRIME, NIXDORF, NORSK DATA, OLIVETTI, HEWLETT PACKARD, UNISYS, STRATUS, NCR, SEQUENT, WANG, APOLLO in SUN su vlasnici navedenih zaštitnih žigova.

INFORMACIJE:
Tovarna meril, RAČUNALNIŠKI INŽENIRING KOPA,
Kidričeva 14, SLOVENJ GRADEC
telex: 33238, telefon: 062-841-798

RAČUNALNIŠKI INŽENIRING — KOPA — KUĆA MUDRIH REŠENJA

NFK NOVKABEL

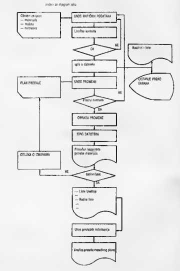
SOFTVERSKI PAKET - PLANIS

U okviru V simpozijuma upravljanja proizvodnjom organizacionim sredstvima koji će se održati 12 do 15 oktobra na zagrebačkom sajmu za vreme održavanja INTERBIRO-a u klubu privrednika — kineskog paviljona biće demonstriran i softverski paket — PLANIS

Programski proizvod PLANIS pruža mogućnost izrade operativnih planova proizvodnje u svim prekidnim procesima Na osnovu plana prodaje i unapred pripremljenih podataka o proizvodima mikroračunar omogućuje balansiranje potreba proračuna kapaciteta, vremena izrade proizvodnje, lansiranje radnog naloga, praćenje realizacije operativnog plana proizvodnje PLANIS je interaktivni proizvod orijentisan prema korisniku, što omogućuje efikasno uvođenje i eksploataciju Dijagram toka je prikazan na slici

Osnovni podaci koje obuhvata PLANIS su
- — podaci o materijalu
- — podaci o poluproizvodima
- — podaci o gotovim proizvodima
- — podaci o mašinama — kapacitetima
- — normativi materijala
- — normativi vremena
- — katalog operacija
- — planski podaci

Svi podaci formirani su u logičke celine iz kojih PLANIS vrši sledeće funkcije
- — održavanje podataka
- — formiranje strukturnih veza
- — unos promena
- — bilansiranje potreba
- — proračun kapaciteta
- — izrada mesečne planske sastavnice
- — lansiranje radnog naloga
- — obrada povratnih informacija iz procesa proizvodnje
- — prikaz (presek) realizacije operativnog plana
- — statistički izveštaji



RZ KOMERCIJALNI POSLOVI
Poslovnica prodaja
Tel. 021/337-255
21000 Novi Sad
Put Novosadskog partizanskog odreda br. 4

Nešto sasvim blisko

U članku „Objektno programiranje" (Računari 36, str. 28—30) objašnjeni su osnovni pojmovi i istorijat razvoja jedne sasvim nove vrste programiranja. Program Smalltalk/V predstavlja, bez sumnje, najpopularniji objektnoorijentisani jezik na računarima IBM PC/XT/AT. Radi se o veoma kvalitetnom ostvarenju, sa oko 2000 naredbi (metoda) i kompletnom radnom okolinom. Uključeni su interpretar, prevodilac, programski editor, prozori, miš, dibager, grafika visoke rezolucije, meniji, prolog, rad sa DOS-om i još mnoga druga mogućnosti.

Po učitavanju *Smalltalk/V* se prikazuje kao slika **(image)** na ekranu. U zavisnosti od stanja slika u trenutku prethodnog izlaska iz *Smalltalk/V*-a, na ekranu će se videti jedan ili više prozora na sivoj pozadini, kao i kursor. Komunikacija sa sistemom odvija se preko menija — pozivaju se tasterom **Del.** Položaj kursora određuje meni koji će se pojaviti. Sistemski meni je uvek na raspolaganju, a pojavljuje se kada je kursor vrhom na sivoj podlozi. Ostali meniji vezani su za prozore, pri čemu se prozor sastoji od zaglavlja i jednog ili više okana **(pane).** Pogledajmo tipične menije.

Meni, meni, meni

Sistemski meni ima sledeće opcije: **dos shell** (odlazak u DOS), **space/speed** (ubrzanje programa na račun prostora i obratno), **exit Smalltalk** (izlazak iz programa), **browse disk** (pregled datoteka na disku), **open workspace** (otvaranje novog prozora), **browse classes** (prelistavanje sadržaja klasa), **redraw screen** (ponovno iscrtavanje ekrana), **save image** (snimanje slike na disk) i **run demo** (izvršavanje demo programa).

Odlazak u DOS ostavlja *Smalltalk/V* pritajenog u memoriji i prazni 128 K za neki drugi program. Samo prvih 37 K *Smalltalk/V-a* mora ostati u memoriji, a sve ostalo se može izbaciti. Opcija **space/speed** jedno je od bitnih obeležja *Smalltalk/V*-a. Kratica /V u imenu znači „virtuelni". *Smalltalk/V* zaobilazi memorijsko ograničenje od 640 K tako što automatski izbacuje nepotrebne objekte na disk, tj. radi sa virtuelnom memorijom. To znači da se u ovom programskom jeziku bez problema mogu praviti programi čija je dužina ograničena samo raspoloživim prostorom na disku. Pomoću **exit Smalltalk** može se završiti rad sa *Smalltalk/V-om*. Tu opet vodimo dijalog sa ukupno tri opcije: **forget image** (izaći, ne snimati sliku na disk), **continue** (vratiti se u *Smalltalk/V*, ništa se ne snima), **save image** (izaći, snimiti kompletno stanje sistema na disk). Prva i treća opcija vraćaju u DOS. Prva se koristi kada treba zaboraviti sve promene od trenutka poslednjeg snimanja, a poslednja snima kompletno stanje — trenutni oblik ekrana kao i sve pridodate naredbe. U sledećem učitavanju *Smalltalk/V-a* pozdravni ekran će biti isti kao i na izlasku, a i sve naredbe će biti tu. *Smalltalk/V* tako raste i postaje bogatiji svaki put kada je izabrana kombinacija **exit Smalltalk/save image.** Nove naredbe postaju time ravnopravne sa prethodno unesenim, sistemskim naredbama.

Opcija **open workspace** zahteva od korisnika da otvori prozor i da mu da ime. Otvaranje prozora vrši se interaktivno: gornje levo i donje desno teme se kursorima ili mišem dovode na željeno mesto na ekranu, potvrdi se njihova lokacija i zatim se navede ime prozora. Ime se smešta u zaglavlje novog prozora. Na ekranu istovremeno može postojati neograničen broj prozora, ali je samo jedan aktivan. Zaglavlje aktivnog prozora je crno, a kod svih ostalih prozora zaglavlja su bela. Prozor se aktivira jednostavnim dovođenjem kursora negde u oblast prozora i pritiskom na levi taster na mišu ili na sivi + taster na tastaturi. Postoji jedan poseban prozor — System Transcript. On je uvek prisutan na ekranu makar samo sveden na zaglavlje. Svi ostali prozori mogu se obrisati, ali ne i on. Obično se u njemu interaktivno testiraju programi.

Opcija **browse disk** služi za interaktivno pregledanje sadržaja datoteka i imenika na disku. Po biranju ove opcije korisnik mora da otvori prozor na gore opisani interaktivni način, pri čemu *Smalltalk/V* sam daje ima **Disk Browser**. Svako od četiri okna ima svoj meni, a zastupljene su uobičajene operacije iz DOS-a: premeštanje u neki drugi imenik, brisanje imenika i u njemu zadržanih datoteka (Ovo i nije standardna operacija DOS-a), otvaranje novog imenika, odabir datoteka, njihovo premeštanje, štampanje, promena atributa, preimenovanje, kopiranje i stvaranje novih datoteka. U donjem oknu može se videti i menjati sadržaj odabrane datoteke. Ukratko, **Disk Browser** omogućava sve česte DOS operacije bez napuštanja *Smalltalk/V-a*.

Opcija **browse classes** na već opisani način vodi u stvaranje prozora **Class Hierarchy Browser.** U gornjem levom — od ukupno pet okana — izložena je kompletna hijerarhija klasa. Biranjem jedne od klasa gornje desno okno popunjava se metodima te klase, biranjem jednog od tih metoda donje — najveće — okno prikazuje tekst metoda. Postoje još dva okna, **class** i **instance**. Ona naznačuju da li se prikazuju metodi klasa (class method) ili metodi vezani za primerke klase (instance method). U svakom slučaju, tekst metoda se može pisati i/ili menjati i prevoditi iz donjeg tekstuelnog okna. **Class Hierarchy Browser** je dakle mesto gde se menjaju, dodaju ili ukidaju čitave klase metoda.

Opcija **save image** iz glavnog menija samo je skraćenica za kombinaciju **exit Smalltalk/save image**. Konačno, **run demo** izvršava već isprogramirane demonstracione programe. Programi su efektni i koriste znatne grafičke i animacijske mogućnosti *Smalltalk/V-a*.

Prozorski meniji

Prozor se sastoji od zaglavlja sa imenom i jednog ili više okana. Meni zaglavlja deluje nad prozorom kao nad celinom. Tu su opcije: **label** (promeni ime prozora), **collapse** (svedi prozor samo na zaglavlje da ne bi zauzimao ekran), **cycle** (napusti ovaj i aktiviraj sledeći prozor), **frame** (pomeri i/ili promeni veličinu prozora), **move** (pomeri prozor), **close** (zatvori tj. uništi prozor). Nemaju svi prozori sve ove opcije, a System Transcript ima samo **cycle, frame** i **move**.

Postoje tri vrste okana: tekstualna, sa listama i grafička, a razlikuju se međusobno po menijima. Da bi se bilo šta uradilo sa tekstom, potrebno ga je označiti. To se radi ili mišem ili kombinacijom sivog tastera + kursora i sivog tastera –. Označeni tekst vidi se kao bela slova na crnoj podlozi. Zatim se poziva meni tekstualnog okna. Standardne opcije su: **copy** (copiranje označenog teksta u bafer), **cut** (isto kao **copy**, samo se označeni tekst odmah i briše), **paste** (tekst iz bafera se ubacuje na mesto kursora), a tu su i opcije **show it, do it, save** i **next menu**. Ova poslednja opcija je uvod u još jedan meni, ovog puta za pretraživanje po zadatom tekstu, globalne zamene i slično. Opcija **save,** zapravo, predstavlja način da se odabrani tekst prevede i snimi na disk, što odgovara opciji **compile** kod klasičnih prevodilaca. Preostale dve opcije **show it** i **do it** su za praktičan rad najvažnije. Sa **do it** se označeni tekst interpretira i izvršava, a **show it** je isto to, samo se još vrednost poslednjeg izračunatog objekta ispisuje na kraju obeleženog teksta. Tako izračunata vrednost je tekst ravnopravan sa ▶

Sistemski zahtevi i verzije programa

Smalltalk/V radi na računarima sa dve disketne jedinice ili sa tvrdim diskom, sa najmanje 512 RAM-a, monohromatskim ili monitorom u boji, jednom od sledećih grafičkih kartica: CGA, EGA, Hercules, AT&T (Olivetti 640×400) i Toshiba T3100, a potreban je i DOS verzija 2.0 ili neka sa većim rednim brojem.

Poželjno je imati svih 640 K RAM-a, Microsoft usaglašenog (kompatibilnog) miša i matematički koprocesor.

Proizvođač je Digitalk Inc., 9841 Airport Blvd, Los Angeles, CA 90045 USA, tel 800-922-8255. Cena je 99 dolara plus poštarina. Trenutna verzija programa je Release 2.00. Upravo je počela da se prodaje verzija sa 32-bitnom internom arhitekturom, tako da se izvršava samo na računarima sa procesorima 80286 i 80386 uz bar jedan megabajt memorije. Za uzvrat, *Smalltalk/V286* (200 dolara) radi u zaštićenom režimu (protected mode) i direktno adresira svu raspoloživu memoriju (sve do 16 MB). Uskoro će biti završen razvoj verzije *Smalltalk-a* za tzv. MaxBoard — hardverski ubrzivač AT računara (ne i XT modela!) do na brzinu od 40 MIPS-a! Digitalk takođe prenosi svoje programe i na makintoš II i SE, kao i operativni sistem OS/2.

ostalim tekstom u prozoru, što znači da su u *Smalltalk/V-u* programi testiraju na veoma jednostavan način. I to sasvim interaktivno. Baš kao i FORTH *Smalltalk/V* je istovremeno i interpretar i prevodilac: sa interpeterom se novi metod testira, pomoću opcija **copy** i **paste** prebacuje u **Disk Browser Window**, tu se opcijom **save** prevodi i trajno postaje deo programskog sistema.

Kao i kod bejzika i ovde se interaktivno mogu menjati vrednosti varijabli i zatim nastaviti izvršenja programa. U slučaju neke greške pojavljuje se prozor sa obaveštanjem, a programer i sam može da naredi aktiviranje dibagera.

Objektno-orijentisano programiranje drastično povećava programersku produktivnost, jer se veći deo naredbi preuzima iz već postojećih klasa i metoda. Združivanje interpretera i prevodioca još više olakšava rad. Novi metod se može interaktivno testirati, programer je skoncentrisan samo na njega, po završetku testiranja metod je proveran i (verovatno) neće u budućnosti unositi probleme u program. A upravo to i jeste najveći dobitak kojeg objektno-orijentisano programiranje nudi programerima: sa povećavanjem programa kompleksnost se **NE** povećava!

Klasa Magnitude — brojevi

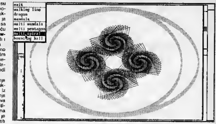
Da bi ceo ovaj pristup funkcionisao od samog početka, *Smalltalk/V* se isporučuje sa 100 klasa i oko 2000 metoda. Klasa **Magnitude** okuplja sve veličina koje se mogu porediti, meriti, uređivati i brojati. Njene glavne podklase su **Association** (pridruživanje dva objekta na osnovu relacija ključ — podatak), **Character** (prošireni, tj. IBM skup znakova), **Date** (datumi), **Time** (časovnik realnog vremena) i **Number** (brojevi raznih vrsta). *Smalltalk/V* dolazi sa sledećim klasama brojeva: **Float** (brojevi sa pokretnim zarezom), **Fraction** (racionalni brojevi) i **Integer** (celi brojevi). U osnovnoj verziji *Smalltalk/V* ne izvodi računske operacije sa brojevima klase **Float** ako u računaru nije instaliran matematički koprocesor. To proširenje može da se nabavi kao dodatni softver za pedesetak dolara. Naravno, uvek se može dokupiti i sam koprocesor za stotinak dolara. A ko neće ni jedno ni drugo, može da proglasi da je klasa **Float** identična klasi **Fraction**, čime se aritmetičke radnje usporavaju ali zato nema računskih grešaka (svi brojevi se tada interno predstavljaju kao količnici parova celih brojeva, npr. 1.5=3/2). Uzgred, *Smalltalk/V* može da predstavi cele brojeve do 65535 bajtova preciznosti.

Ilustracije radi, navodimo neka od ukupno 27 metoda klase Time (vreme): prekid interapta sata realnog vremena, omogućavanja tog istog interapta, konverzija trenutnog vremena u niz bajtova, vreme i datum u ovom trenutku, izračunavanje trajanja izvršenja nekog bloka, usklađivanja rada sata i pokreta miša, trenutno vreme, ukupan broj milisekundi, sekundi, minuta, sati od početka dana do trenutnog vremena, razna poređenja vremena u sekundama, štampanje vremena u ASCII obliku i tako dalje.

Klase toka — datoteke

Rad sa datotekama omogućen je klasama toka **(stream classes)**. Koriste se za pristupanje datotekama, logičkim i fizičkim uređajima, kao i objektima koji su predstavljeni kao nizovi znakova.

Najvažnije podklase klase **Stream** (tok)



su: **ReadStream** (samo učitavanja), **WriteStream** (samo pisanje), **ReadWriteStream** (i čitanja i pisanja istovremeno), **FileStream** (rad sa datotekama), uz još neke. Svaka klasa sadrži interni pokazivač na mesto čitanja ili pisanja.

Evo nekoliko poruka u vezi sa ovim klasama. Sa objekta klase **ReadStream** sledeći znak se čita porukom **next**, i piše na objekat klase **WriteStream** porukom **nextPut**. Poruka **atEnd** otkriva kraj objekta klase **ReadStream**. Sadržaj se upisuje sa porukom **contents**. Za položaj pokazivača mesta čitanja/upisa može se upitati porukom **position**, ili sa sličnom porukom **position: anInteger** pokazivač namešta na željeni položaj. Poruka **nextPutAll: aCollection** upisuje elemente kolekcije u tok. Upisivanje može biti i direktno, porukom poput **next: anInteger put: anObject.**

> **Instaliranje**
>
> Dokumentacija ima 514 strana i može da posluži i kao udžbenik jezika i kao priručnik. Isporučuju sa ukupno tri diskete i nisu zaštićene od kopiranja. Instaliranje na tvrdi disk izvodi se iz dva koraka. Prvo se otvori imenik (directory) i kopira sadržaj svih triju disketa obično **copy *.*** naredbom iz DOS-a. Posle toga se naredbom
>
> **install hard (grafička kartica)**
>
> takođe iz DOS-a, saopštava jedna od spomenutih grafičkih kartica i — *Smalltalk/V* je spreman za rad. Ako bi umesto **hard** stajalo **soft** u gornjoj naredbi, *Smalltalk/V* bi bio spreman da radi sa dve disketne jedinice. Sve u svemu, instaliranje *Smalltalk/V-a* prilično je jednostavno.

Klasa **FileStream** je glavni interfejs prema DOS-u. Za nju važe sva gore spomenute poruke. Sledeći red deklariše novu promenljivu **poglav** kao datoteku sa imenom poglav.dat:

```
| poglav |
poglav := File pathName: 'c:/word/poglav.dat'
```

Imena varijabli navode se između dva uspravne crte. Poruka pathName istovremeno i kreira datoteku i imenuje je. Evo još jednog primera. Sledeće naredbe pretražuju zadati ulaz i svako pojavljivanje znaka % zamenjuju današnjim datumom.

```
"Zamena znaka % dnevnim datumom."
| ulaz izlaz znak današnji Datum |
današnji Datum := Date today printString.
Ulaz := ReadStream on: 'Današnji datum je %'.
Izlaz := WriteStream on: String new.
[ ulaz atEnd ]
    whileFalse: [
        (znak := ulaz next)=$%
            ifTrue: [izlaz nextPutAll: današnji
            Datum]
            ifFalse: [izlaz nextPut: znak]].
            izlaz contents
```

Tekst između dvostrukih navodnika je komentar, a između jednostrukih je string. Varijabla **ulaz** je tok nad stringom tako da naredbe **ulaz next** daje znak po znak iz stringa '**Današnji datum je %**'. Dolar ispred znaka % označava da se uzima znak sam sa sebe (a ne, recimo, ASCII vrednost). Varijabla **izlaz** je tok po stringu koji se upravo oformio dejstvom poruke **new** na globalnu varijablu String. Petlja **whileFalse:** iscrpljuje string **ulaz** znak po znak, sve do kraja stringa, ispituje da li je učitani znak jednak %, i deluje u skladu sa ishodom pitalice. Konačno, poruka **contents** vraća string koji sadrži sva objekta upisana na **izlaz.**

Programi u *Smalltalk-u* imaju jednu izuzetnu osobinu: zavise samo od ponašanja, tj. od algoritma a ne od strukture podataka, to znači da se programi mogu pisati i testirati prvo sa brzim podacima u memoriji, a da se zatim jednostavno prelazi na spoljašnje datoteke. U gornjem metodu dovoljno je staviti nova definicija za tokova **ulaz** i **izlaz**, recimo ovako:

```
ulaz := File namePath: 'c:/smaltalk/glava1.dat'
izlaz := File namePath: 'c:/smaltalk/glava2.dat'
```

pa da program počne da radi sa realnim podacima.

Tokovi ne moraju čitati ili pisati samo po stringovima. Evo primera sa nizovima:

```
"Izračunavanje nekih faktorijela."
 | ulaz izlaz n |
ulaz := ReadStream on: #(1 5 10 20).
Izlaz := WriteStream on: Array new.
[ulaz atEnd]
    whileFalse: [
        izlaz nextPut: ulaz next factorial].
Izlaz contents
```

Povisilica # označava početak niza. Gornji metod uzima broj iz toka **ulaz**, izračunava njegov faktorijel i smešta ga u tok **izlaz.** Na sličan način bi se organizovalo štampanje na štampaču, koji ne bi bio ništa drugo do jedna vrsta toka znakova.

Kolekcije — rečnici, vreće, nizovi

Pod kolekcijom se podrazumeva grupa srodnih objekata. Klasa **Collection** definisan je kao apstraktna da bi objedinila protokola za pregledanje pronalaženje dodavanje i promenu članova kolekcije. Kolekciju čine sledeća četiri atributa: (1) jasna definicija relacije uređenosti u odnosu na članova kolekcije, (2) da li je dužina kolekcije stalne ili promenljiva, (3) da li su dozvoljani duplikati članova kolekcije i (4) da li se do pojedinih članova kolekcije može doći preko skupa nekih ključeva (npr. indeksa ili pretraživanjem ključeva). *Smalltalk/V* poznaje sledeće podklase klase **Collection**

```
Collection
    Bag
    IndexedCollection
        FixedSizeCollection
            Array
            ByteArray
            Interval
            String
                Simbol
        OrderedCollection
            SortedCollection
    Set
        Dictionary
            IdentityDictionary
```

Svaka kolekcije se može pretvoriti u **Array** (niz), **Bag** (vreću), **Set** (skup), **OrderedCollection** (uređenu kolekciju) ili **SortedCollection** (sortiranu kolekciju).

Neka od ovih klasa sasvim su slične odgovarajućim strukturama podataka u paskalu, npr. **Array, String, ByteArray,** ali neka mogu biti kompletno iznenađenje za paskal programere. Takva je klasa **Bag** (vreća). U njoj su duplikati dozvoljeni, a uređenje elemenata je proizvoljno. Nema spoljašnjih ključeva (npr. indeksa), tako da standardne poruka **at** i **at: put** ostoju bit odgovore. Vreće se obično upotrebljavaju za brojenje frekvencije. Sledeći primer izračunava frekvenciju slova u ulaznom redu

```
:s f·
S =Prompter prompt  Unesi red  default' '.
f :=Bag new
s do  [:c I isLetter I fTrue  [f add  c asLowerCase]].
f
```

Klasa **Set** je isto što i **Bag,** osim što višestruki elementi nisu dozvoljeni. Sledeće naredbe sortiraju spisak reči u datoteci

```
ulaz reči reč
ulaz :=File pathName  in.fil
reči :=Set new
[(reč:=ulaz nextWord) isNil]
    whileFalse  [reči add  reč asLowerCase].
reči asSortedCollection
```

Poruka **nextWord** učitava **reč** iz datoteke **in.fil**. Svaka reč se pretvara u mala slova, smešta u skup **reči,** i u poslednjoj liniji sortira konvertovanjem u kolekciju čiji su članovi sortirani.

Članovi klase **Array** su uobičajeni nizovi. Zahvaljujući tzv. kasnom povezivanju **(late binding),** nizovi (kao i sve ostale kolekcije) mogu da sadrže bilo koje objekte. Evo primera

```
array skup
skup :=Set new
array  Array new  4.
array :=1 put  1
array at  2 put  skup
array at  3 put  Ovo je neki string
array at. 4 put  #(12 13 24 45).
array :=5
```

Prvi član niza **array** je broj 1, drugi je skup, treći je string, četvrti je opet neki konstantni niz — toliko raznorodnih tipova i svi članovi jednog niza! Konačno, istu varijablu **array,** proglašavamo za ceo broj. Sve je to u klasičnim programskim jezicima apsolutno nezamislivo — baš kao i nove vrste petlji koje slede

Neke nove petlje

Naredbi FOR ... TO ... DO iz paskala odgovara naredbi **to do** i ona je već opisana u spomenutom članku. Osim nje postoji i jednostavnija naredba **timesRepeat**. Ona ponavlja blok izvestan broj puta

```
Index
index :=0
3 timesRepeat  [index :=index+1]
```

Smalltalk/V nudi još tri vrste petlji. Prva je **select**. Ona iterira svaki deo primaoca i vraća sve elemente za koji se blok izračunava kao **true** (tj. kao tačan). Na primer

```
„Izdvajanje neparnih brojeva iz niza"
#(1 2 3 4 5 6 7 8 9 10) select  [ c I c odd]
```

i rezultat je niz #(1 3 5 7 9)

Slična je poruka **reject** — njome se vraćaju samo delovi primaoca za koje se vrednost bloka izračunava kao **false** (logički uslov nije ispunjen). Na primer

```
„Izdvajanje neparnih brojeva iz niza"
#(1 2 3 4 5 6 7 8 9 10) reject  [ c I c odd]
```

i rezultat je sada niz #(2 4 6 8 10)

Poslednja petlja je **collect**. Ona izračunava i vraća svaki deo primaoca bezuslovno. Na primer

```
„Izračunavanje kvadrata svih elemenata niza"
#(1 2 3 4) collect  [ n I n* n]
```

vraća rezultat #(1 4 9 16)

Na sledećem jednostavnom primeru pokazaćemo bitnu razliku između *Smalltalk/V*-a i npr. paskala. Zadatak glasi: svaki element niza povećati za jedan. Po ugledu na pascal, mogli bi se i na *Smalltalk-u* napisati nešto poput ovoga

```
niz
niz :=#(1 3 6 8 8).
1 to  (znakovi size) do  [ I I
    niz at  I put  ((niz at  I)+1)]
```

Koristeći **collect**, pišemo elegantno

```
#(1 3 6 6 6) collect  [ I I  :=I+1]
```

U drugom slučaju se bloku iza **collect** predaje član niza, a ne njegov indeks.

Tu se pomalja još jedna bitna konceptualna različitost. Na paskalu smo primorani da razmišljamo u terminima bajtova odnosno elemenata nizova. U *Smalltalk-u* razmišljamo u terminima nizova kao celine — ulazni rezultat je niz, baš kao i izlazni. *Smalltalk* programer je pošteđen stalnog spuštanja na nivo hardvera — iako mu ništa ne stoji na putu da programira i tako ako hoće.

Grafika i prozori

Bez ikakvih pretenzija na potpunost spomenućemo da *Smalltalk/V* podržava veoma bogat skup poruka za crtanje i animaciju. Mogu se eksplicitno definisati tačke, pravougaonici, elipse i slični oblici. Pri tome se koristi tzv. BitBlt grafika, što znači da se čitavi delovi memorije veoma brzo i odjednom premeštaju na ekran. Pošto su i slova zapravo samo bit-po-bit iz memorije preslikana grafika, oblici i veličina slova mogu se menjati po volji. Različiti oblici slova istovremeno mogu biti na ekranu, tako da nije čudo što su i neki komercijalni programi za stono izdavaštvo rađeni baš u ovom jeziku.

Animacija se izvodi veoma jednostavno, ali postoji opasnost da se stanje sistema „prepuni" objektima za animaciju i tada se program usporava zbog prebacivanja objekata na disk.

Razvoj novog interaktivnog programa zahteva temeljito poznavanje tehnologije rada sa prozorima, što je detaljno obrađeno u priručniku.

Tako blisko čoveku

Objektno-orijentisano programiranje postaje sve značajniji pokret. Krajem prošle godine održana je sa velikim uspehom druga svetska konferencija na tu temu, sa preko 1100 učesnika. Toliki interes je lako shvatljiv. Objektno programiranje je nova šansa za širenje računara, jer se zahvaljujući njemu i obični korisnici odvažuju na programiranje. *Smalltalk/V* pruža šansu za razvoj potpuno novih programa — programe koji su nekom zaista potrebni, ali nema dovoljno kupaca da bi se komercijalna eksploatacija isplatila ili takav softver može da razvije samo vrhunski stručnjak za datu oblast. Najveći deo obrazovnog softvera ispunjava oba ova kriterijuma, pa nije ni čudo što ga tako malo ima — i kod nas i na Zapadu.

Prve laste već su tu: interaktivni udžbenik fiziologije u Vankuveru, upravljanje NASA-inim komunikacionim satelitom, simulacije populacije belorepih jelena u Džordžiji, stono izdavaštvo u Arizoni, simulacija neuronskih mreža, rekonstrukcija puteva na Floridi, razvoj grafičkih jezika u IBM-u, generisanje izveštaja o pacijentima na univerzitetu UCLA u Kaliforniji, čak 20 puta ubrzano prikupljanje podataka od stranaca u Francuskoj, simulacija mikro-tržišta na ekonomskom fakultetu u Pitsburgu ... I što je najvažnije, većinu tih programa pravili su inženjeri, doktori, predavači, dakle koji nikada nisu učili programeri. *Smalltalk*, *Actor* i drugi objektno-orijentisani jezici odslikavaju svet u računaru baš kao što se svet odslikava u čovekovom umu i — pomalo paradoksalno — čine računare bližim čoveku upravo zato što uklanjaju masu tehničkih detalja.

Duško Savić

Dodatni programi

Uz osnovnu verziju *Smalltalk/V*-a dobija se i program za 2D crtanje mišem. Tek je nešto slabiji od poznatog programa *Dr. Hallo* — a najveća mu je vrednost što svaki crtež može postati objekat i kao takav se upotrebljavati u *Smalltalk/V* programima.

Iako je u priručniku jedva spomenuto, *Smalltalk/V* je još i bolja vrednost nego na prvi pogled. Uz njega se dobija i kompletan prevodilac za prolog i to tako da se naredbe iz *Smalltalk/V*-a i Prolog-a mogu veoma lako koristiti zajedno.

Postoji i treći program demonstrativnog i edukativnog karaktera.

U koži superjuzera

Da bi jedan višekorisnički računar uopšte mogao da funkcioniše, mora postojati takozvani upravnik sistema (system manager ili, u terminologiji juniksa, superuser) — ovom korisniku je dato pravo da uključuje i isključuje računar, analizira, menja i briše svaku datoteku, utiče na izvršavanje svih procesa i, naravno, da autorizuje druge korisnike, omogućavajući im tako da normalno koriste sistem. U ovom, završnom, poglavlju naše male škole juniksa upoznaćemo naredbe rezervisane za upravnika sistema.

Juniks računar prepoznaje upravnika po korisničkom imenu — privilegije ima isključivo korisnik koji se „zove" *root*. Ukoliko dakle želite da upoznate ovlašćenja superkorisnika, moraćete da se raspitate o lozinki korisnika *root* (na većini juniks sistema početna lozinka je takođe *root*, ali je, međutim, prilika da ju je upravnik centra odmah promenio) i da tu lozinku unesete u toku početne komunikacije sa računarom. Alternativno, na nekim verzijama juniksa uvedena je komanda *su* (od *SUperuser*) koju svaki korisnik može da otkuca u toku normalnog interaktivnog rada — računar još zahteva kucanje *root* lozinke i pretvara običnog korisnika u *superuser*-a!

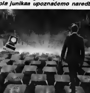
Upravničke privilegije

Pre nego što nastavimo sa opisom potencijalno destruktivnih naredbi, dužni smo da ponovimo upozorenja koja redovno navode razni juniks priručnici. Prilikom izvršavanja naredbi koje zadaje *superuser*, broj provera koje operativni sistem izvršava je minimalan i svodi se na kontrolu sintakse — računar ne proverava da li će izvršavanjem komande biti oštećen neki drugi korisnik ili čak sam operativni sistem, odnosno prateći hardver. Nepažljivim baratanjem komandama koje ćemo opisati ugrožavate integritet sistema, što znači da nije teško dovesti do problema koja mogu da reše jedino serviseri.

Brinući o ovakvim posledicama, juniks će vas stalno upozoravati da ste *superuser* i pretvoriti standardni prompt (dolar) u neki specijalni znak, na primer, tarabu (#). Upravnik sistema u svakom slučaju treba da ima i normalno korisničko ime — pod imenom *root* prijavljivaćemo se samo kada treba da izvršimo neke operacije nedostupne običnom korisniku.

Izrazito loša strana juniksa je što postoji samo jedan privilegovani korisnik (*root*), što znači da, ukoliko računski centar ima nekoliko ovlašćenih superkorisnika (neophodno da bi, na primer, u računskom centru koji radi 20 ili 24 časa dnevno stalno dežurao upravnik koji može da se nađe na usluzi ostalim korisnicima), više ljudi mora da zna *root* lozinku. Ukoliko se u nekom trenutku u sistemu dese neke neželjene stvari, vrlo je teško utvrditi koji ih je od privilegovanih korisnika izazvao (ne treba, naravno, zanemariti ni mogućnost da je neko od hakera ili nastrojenih običnih korisnika pogodio loše izabranu lozinku). Videćemo da na nekim drugim operativnim sistemima, kao što je VMS, i drugi korisnici mogu da budu ovlašćeni da, koristeći svoja standardna imena, obavljaju precizno određeni deo funkcija *superuser*-a.

Upravnik pri redovnom radu koristi razne programe koji su običnim korisnicima nepotrebni i koji im, dakle, mogu biti i nedostupni (sećamo se da se naredbom *chmod* određenim klasama ili svim korisnicima može zabraniti čitanje, izmena ili izvršavanje pojedinih programa, odnosno datoteka). Ti se programi uglavnom nalaze u direktorijumima (etc) bin i (usr) bin, što znači da komandu PATH treba podesiti tako da pretražuje i te kataloge. Ukoliko koristimo *Bourne Shell*, otkucaćemo nešto poput **PATH=(etc:$PATH, export PATH** dok odgovarajuća komanda za *C Shell* (videćemo da mnoge komande koje koristi *superuser* značajno variraju na raznim verzijama juniksa) glasi **set path=(/etc $path).**



Uključivanje sistema

Retki su računarski sistemi koji rade 24 sata na dan — većinu računara svakodnevno treba uključivati (engleski izrazi su *bootstrapping* ili prosto *booting*, domaći programeri uglavnom govore o „dizanju sistema") i isključivati. Samo „okretanje prekidača" jasno zavisi jedino od hardvera kojim smo se opremili, softverski deo „dizanja sistema" je za razne juniks mašine praktično identičan.

Posle uključivanja, računar iz ROM-a učitava neophodne programe za testiranje, proverava razne komponente i na kraju se javlja kao **jedno**korisnički sistem koji opšti isključivo sa konzolnim terminalom na kome je automatski prijavljen korisnik *root*. Prva operacija koju ćemo svakako obaviti je podešavanje vremena — računar „misli" da se vreme nije promenilo od trenutka kada je isključen. Otkucaćemo, dakle, nešto poput *date 880329232530*, što znači da je tekući datum 29. mart 1988. i da časovnik treba da pokazuje 23 časa, 25 minuta i 30 sekundi. Na Bell System III i V Unix-u format komande *date* je nešto drugačiji — u našem bi slučaju trebalo otkucati *date 0329232588*. Bilo kako bilo, računar će proveriti ispravnost vremena i ispisati datum u standardnom formatu; biće ispisan i dan u nedelji koji je, bez potrebe da ga specificiramo, izračunat iz datuma.

Sledeća operacija je proveravanje konsistentnosti podataka na disku — ukoliko sistem nije ispravno isključen, moguće je da su neki podaci nepovezani sa matičnim datotekama. Proveru i ispravljanje pronađenih grešaka vrši program *fsck* — obično ga startujemo bez parametara, a računar opisuje uslove koji se proveravaju i postavlja pitanja o tome treba li ispravljati svaku od pronađenih nedoslednosti (namerno ne kažemo *greški* jer je moguće da se radi o sasvim ispravnim datotekama koje hardver usled kvara koji će moći da otklone samo serviseri nepravilno prepoznaje). Detaljna upotreba programa *fsck* je svakako opisana u dokumentaciji vašeg sistema — radi se o rutini koja je u značajnoj meri mašinski zavisna.

Pošto su provere završene, pritisnućemo CTRL D (na Bell System III i V pre toga treba otkucati *init* 2) i računar će preći u višekorisnički mod — svi priključeni terminali su spremni da komuniciraju sa korisnicima. Pri tome se automatski izvršava procedura isporučena uz sam računar koja podešava razne parametre i aktivira procese neophodne za komforan rad — spuler za štampač, časovnik i tome slično. Ovu proceduru ćete, kada bolje upoznate juniks, moći i sami da dopunite prilagođavajući tako sistem svojim aplikacijama.

Datoteka *password*

Pošto je sistem aktiviran, može da mu pristupa bilo koji autorizovani korisnik. Na početku ove male serije tekstova rekli smo da nove korisnike autorizuje upravnik računskog centra koji im dodeljuje korisnička imena i početne lozinke. Sada ćemo videti kako se to dodeljivanje obavlja.

Datoteka *passwd* u direktorijumu */etc* sadrži relevantne parametre o svim korisnicima sistema. Na slici 1 vidimo mogući sadržaj datoteke *passwd* na nekoj manjoj juniks mašini.

Svaki red odgovara jednom od korisnika i sadrži nekoliko polja koja su odvojena dvotačkama. Prvo polje je korisničko ime (npr. *root* ili *demo*), sledi lozinka kodirana na način o kome ćemo tek govoriti, identifikacioni kod korisnika, njegova grupa, komentar, početni katalog i ime takozvane *login* procedure. Svako od ovih polja ćemo detaljno upoznati.

Korisničko ime je niz od najviše osam slova i cifara — uobičajeno je da korisničko ime predstavlja neku vrstu skraćenice od imena, prezimena i funkcije korisnika.

Lozinka je potencijalno najinteresantnije polje. Najjednostavnije bi bilo da se u datoteku *passwd* upiše sama lozinka koju je upravnik sistema dodelio ili korisnik izabrao, ali bi ovakvo rešenje bilo prilično nepraktično: svaki bi korisnik mogao da otkuca *cat /etc/passwd*, pročita lozinku *root* korisnika, prijavi se sistemu kao *superuser* i napravi svakakvu štetu. Zato su lozinke šifrovane na način koji zavisi od implementacije juniksa — šifrat je tekst od najviše 13 znakova među kojima se nalaze velika i mala slova, cifre, tačke i kose crte. Kada korisnik otkuca lozinku, računar je šifruje i proverava da li dobijeni šifrat odgovara sadržaju datoteke *passwd* — integralni tekst lozinke se nigde ne „pamti", što znači da čak ni *superuser* ne može da sazna lozinku nekog korisnika!

Lozinka se postavlja **isključivo** pomoću komande *passwd*, premda bi *superuser* teorijski mogao da je promeni i posredstvom bilo kog standardnog editora teksta. Nevolja je u tome što se nasumičnom promenom teksta lozinke praktično onemogućava korišćenje odgovarajućeg korisničkog imena — možda će posle ove promene i postojati lozinka koja bi „otključala" datog korisnika, ali će ona verovatno predstavljati neki besmislen niz slova koji je praktično

nemoguće pogoditi. Čak ni kopiranje lozinke jednog korisnika iza imena drugog ne daje smislene rezultate: prilikom šifrovanja koristi se i samo korisničko ime što znači da čak ni kodirane lozinke korisnika koji su izabrali istu šifru neće biti jednake! Korišćenjem editora moguće je samo ukloniti lozinku — ako obrišemo svih 13 znakova računar pri prijavljivanju korisnika neće zahtevati nikakvu lozinku. U našem je primeru to iskorišćeno kod korisnika *igre*.

Koliko je ovakav sistem kodiranih lozinki siguran? Ne naročito — analizom programa *passwd* čiji izvorni oblik u principu ne predstavlja posebnu tajnu mogao bi se saznati algoritam koji dati juniks koristi za kodiranje lozinki. Čak i ako inverzna transformacija ne postoji, mogao bi se napisati program koji će posle određenog broja iteracija otkriti neku od (smislenih ili besmislenih) lozinki koje „otključavaju" račun zvani *root* — dalje „provala u sistem" ne predstavlja poseban problem. Sve zlo potiče od činjenice da je da bi juniks normalno funkcionisao potrebno omogućiti svakom korisniku da čita sadržaj datoteke *passwd* — videćemo da na nekim drugim operativnim sistemima kao što je VMS datoteka sa lozinkama potpuno nepristupačna običnom korisniku!

Identifikacioni kod korisnika je jednostavno broj između 1 i 65 535 — potrebno je da se svakom korisniku dodeli jedinstven broj uz pomoć koga će se tačno znati kome pripada koja datoteka. Identifikacioni broj 0 rezervisan je za korisnike *root*.

```
# cat /etc/passwd

root.FvcmBwshWQSZ.:0:1.Superuser:/:/bin/csh
bin:yqPGysMHdmZSy:1.1:Commands:/bin:/bin/csh
gasi:EriciiXwMnswK·2:1:Shutdown./bin:/bin/sync
dejan.taycGG/uBtcTV.3.2:D. Ristanovic./usr/dejan:/bin/csh
d1:QphAhhoxRRGbe:4:2.Probni korisnik:/usr/dejan./bin/csh
igre::5·3:Igre./usr/games·/bin/rsh
demo.PESjcrzmndGyA.6·4.Demo sistema./demo./bin/csh
demo1:fUFAdADkGWemZ:7:4.Demo sistema:/demo/print./bin/csh
help·hEcnhXbqPQj8q.8:1·Pomoc korisniku./usr/man·/bin/man
badguy·*************:9:2:Ugasen:/usr/badguy./bin/csh
novi:fWRHIniOERMGP:10:1:Dodavanje korisnika:/usr/etc:/bin/csh
```

Grupa obezbeđuje većem broju korisnika da rade na istom projektu. Opisujući zaštitu prava pristupa pojedinim datotekama videli smo da se na primer grupi korisnika može dopustiti da menjaju datoteku koju ostali korisnici („svet") mogu samo da čitaju. Polje **grupa** definiše grupu kojoj korisnik pripada — *demo* i *demo 1* sa slike 1 pripadaju istoj grupi (20) što znači da mogu zajednički pristupati datotekama koje su za ostale korisnike zatvorene. Posebno je zgodno što se korisnicima čija grupa odgovara *root*-u mogu otvoriti neke sistemske datoteke što znači da oni mogu da obavljaju neke od funkcija *superuser*-a. To je na slici iskorišćeno kod korisnika *gasi* koji, primera radi, može samo da isključi sistem.

Juniks obezbeđuje nekom korisniku da pripada raznim grupama, za šta je zadužena datoteka */etc/group* i komanda *newgrp*; ovim se mogućnostima nećemo baviti pošto su na mnogim sistemima implementirane samo delimično.

Komentar je zapravo kratak „opis" korisnika, na primer njegovo ime i prezime ili status.

Početni katalog je puno ime direktorijuma koji je dodeljen korisniku. Obično se direktorijumi svih korisnika nalaze u katalogu */usr* ili */user*.

Ime login procedure je izuzetno značajno: od takozvane „školjke" *(shell)* zavisi šta će korisnik moći da radi! Na slici 1 „normalnim" korisnicima je dodeljen */etc/csh* odnosno *C Shell* — ova „školjka" obezbeđuje standardnu komunikaciju sa juniksom. Korisniku *igre* je međutim dodeljena procedura *rsh (Restricted SHell)* koja se isporučuje uz Bell Systems III i V i zabranjuje promenu radnog direktorijuma i mnoge druge operacije — korisniku *igre* je dopušteno jedino startovanje programa u osnovnom katalogu tj. igranje. Zanimljive slučajeve predstavljaju i korisnici *help* i *gasi* čija je *login* procedura */usr/man* odnosno */bin/sync* — prijavljivanje na neko od ovih imena jednostavno izvršava specificirani program (pristup HELP biblioteci sistema odnosno programu koji se koristi pri isključenju sistema) posle čega se dijalog sa korisnikom automatski prekida — kao da je otkucano *logout*.

Ime *login* procedure u okviru datoteke *passwd* ne treba mešati sa *login* odnosno *profile* procedurom koju eventualno definiše svaki korisnik. Većina „školjki" omogućava korisniku da u svom osnovnom katalogu kreira datoteku koja se zove *profile (Bourne Shell)* ili *login (C Shell)* i izvršava kada god korisnik započne komunikaciju sa sistemom. U ovu datoteku treba upisati naredbe koje ispisuju pozdravnu poruku i definišu okolinu koja odgovara samom korisniku — možda treba promeniti PATH, uvesti nove komande i tome slično.

Novi i stari korisnici

Pošto smo upoznali datoteku *passwd* pokušaćemo da kreiramo novog korisnika. Ime će mu biti *računari*, identifikacioni broj 50, grupa 20, početni katalog */usr/račun* a *login* školjka */bin/csh*. Pre svega ćemo preći u katalog */etc* (cd /etc) i napraviti rezervnu kopiju datoteke *passwd (cp passwd passwd.old)* koja će poslužiti ako nešto pođe po zlu. Zatim aktiviramo neki editor, na primer *ed*: *ed passwd*. Zahtevamo umetanje teksta na kraj datoteke (komanda a) i kucamo *racunari::50:20:novi korisnik:/usr/racun:/bin/csh*. U sledećem redu kucamo samo tačku, vraćamo se u komandni mod editora i sa w i q upisujemo datoteku na disk i napuštamo editor.

Korisnik *računari* još nije spreman za rad — treba kreirati njegov osnovni katalog i zadati lozinku. Što se početnog kataloga tiče, otkucaćemo *mkdir /usr/racun* dok lozinku dodeljujemo sa *passwd računari*; sistem zatim zahteva da dva puta otkucamo novu lozinku. Zanimljivo je da komandu *passwd* može koristiti svaki korisnik kako bi promenio sopstvenu lozinku ali samo *superuser* može da menja lozinke drugih korisnika!

Ako ste primetili da čitavoj ovoj proceduri nedostaje jedna sitnica, zaslužujete sve čestitke. Kada smo, naime, kreirali početni katalog korisnika (otkucali smo *mkdir /usr/racun*) sistem je zabeležio da je vlasnik „datoteke" (Unix ne pravi naročite razlike između datoteka, uređaja i direktorijuma) */usr/racun* korisnik *root*. Siroti korisnik *računari* nema nikakva prava nad svojim osnovnim katalogom što znači da po svoj prilici neće moći čak ni da kreira datoteke! Zato treba otkucati *chown racunari /usr/racun* a zatim i *chgrp 20 /usr/racun* i tako preneti vlasništvo na korisnika *računari* i grupu 20. Uzdržavanjem od ovog „pokiona" kreiramo korisnike koji imaju ograničena prava da izvršavaju programe koje ne smeju da menjaju, dopunjavaju ili brišu.

Brisanje korisnika je mnogo jednostavnije — uz pomoć editora uklanjamo odgovarajući red iz datoteke *passwd*. Razni juniks priručnici tvrde da je ovakva operacija previše radikalna — na diskovima ostaju datoteke koje ne pripadaju nikome. Zato se preporučuje da polje lozinka upišemo neke nedozvoljene znake, na primer zvezdice. Korisnik i dalje formalno postoji ali ne može da komunicira sa sistemom pošto njegova lozinka jednostavno ne postoji!

Ostale dužnosti

Osim autorizacije novih korisnika, upravnik sistema mora da obavlja i mnoge druge poslove. Verovatno je najbitnija briga o diskovima i datotekama: sistemu mora stalno biti raspoloživ određeni prostor na spoljnim medijima. Zato komandom *df (Disk Free)* treba s vremena na vreme proveravati zauzeće diska. Ako se prazan prostor opasno smanjio, otkucaćemo */etc/quot* i dobiti spisak korisnika sa podatkom o prostoru koji je svaki od njih zauzeo. Posredstvom „elektronske pošte" možemo da upozorimo neskromne korisnike da obrišu neke svoje datoteke, a u ekstremnim slučajevima sledi nasilno brisanje: *superuser* može da obriše čak i datoteke koje mu ne pripadaju! Upravnik sistema, jasno, treba da se brine i o redovnom *backup*-u, sigurnosnom prenošenju sadržaja diskova na druge magnetne medije.

Ponekad ćemo poželeti da proverimo šta se sve događa u sistemu, tj. koji je korisnik potrošio koliko resursa. Za to je zadužena nova verzija komande *who* — kucamo *who /usr/adm/wtmp* (Bell Version 7 i System 3) odnosno *who /etc/wtmp* (Bell System V) i dobijamo važeću „istoriju" rada sistema — ubeleženi su čak i trenuci kada je časovnik doterivan!

Ostaje još da pomenemo dužnost upravnika da dokaže svoju duhovitost redovno menjajući sadržaj datoteke */etc/motd* — u nju je upisana poruka kojom sistem pozdravlja svakog novog korisnika!

Gašenje sistema

Posle napornog radnog dana sistem treba propisno isključiti — prost prekid napajanja izaziva nepravilno zatvorene datoteke i mnogobrojne probleme pri sutrašnjem uključenju. Prvi korak je kucanje komande *who* i, u slučaju da neki korisnici još rade, slanje obaveštenja o kraju radnog vremena. Poruku šaljemo sa *wall* posle čega kucamo tekst i pritiskamo CTRL D. Ako neki korisnici ni posle poruke ne prestanu sa radom, „ubijamo" ih sa *kill-9* ili *killall*. Komandom *ps -ax* odnosno *ps -e* možemo da se uverimo da je aktivan jedino naš tekući proces a onda na Bell System-u III ili V kucamo *init 1* i tako pretvaramo računar u jednokorisničku mašinu. Sledi **veoma značajno** kucanje komande *sync* koja prepisuje sve relevantne tabele iz memorije na diskove — čak i ako sistem gasimo u žurbi („obaramo" ga, što bi rekli programeri) pa nemamo vremena da regularno „ubijamo" korisnike, komandu *sync* svakako treba otkucati. Posle toga prekidamo napajanje na način koji zavisi od karakteristika hardvera kojim smo se opremili.

Ovim „gašenjem sistema" došli smo i do kraja naše male serije napisa o juniksu. Videli smo da je juniks moderan operativni sistem prilagođen ponajviše programerima koji mogu u potpunosti da definišu svoj „radni prostor", i pišući različite „školjke", predizajniraju parametre tako da odgovaraju njihovim potrebama a da istovremeno ove promene ni na koji način ne zasmetaju drugim korisnicima. Loše strane juniksa se, pogotovu kada se radi o početnicima, svode na nesrećno izabrana imena komandi, više nego bedne standardne editore teksta i prilično slabo rešen problem bezbednosti sistema. Videli smo osim toga da juniks nije potpuno standardizovan, što znači da se na raznim verzijama neke komande i pogotovu opcije koriste na različite načine, što početniku svakako ne pomaže. Počevši od sledećih „Računara" predstavićemo još jedan „veliki" operativni sistem koji se zove VMS. Na kraju ćemo biti u prilici da ga uporedimo sa juniksom i spekulišemo o budućnosti operativnih sistema za mikro, mini i velike računare.

Dejan Ristanović

Prelom ili lom

U petom nastavku serije o stonom izdavaštvu, konačno stižemo do rada sa programom za prelom Xerox Ventura Publisher. U ovom nastavku nam je cilj da, na sličan način kao i do sada, naglasimo važnije karakteristike i mogućnosti programa. Detaljno uputstvo za rad planiramo da objavimo na kraju ove serije kao specijalan dodatak „Računara". Ako niste potcenili naše insistiranje da se što veći deo obeležavanja teksta obavi van Venture, ostaje vam samo nekoliko „klik, klik" na tasteru miša i... knjiga je spremna za štampanje! Iako vam se u praksi stvarno može desiti da neke poslove završite privedete kraju sa samo nekoliko poteza, probaćemo da na malo tipičnijem slučaju pokažemo šta je, odnosno šta bi po našem mišljenju trebalo da bude, prava uloga ovog programa.

Da bismo pokazali kako radi porogram za prelom neophodno je da se prvo pozabavimo elementima koji su nužni da bi se do kraja definisao konačni oblik jedne stranice. Do sada smo uglavnom govorili o elementima koji određuju pojavu redova teksta i njihovog uzajamnog odnosa (tip i veličina pisma, razmak između redova, razmak između pasusa, mesto preloma reči na kraju reda itd.). Kada ovako definisan tekst treba položiti na čistu stranicu papira pojavljuje se potreba za dodatnim definicijama. Kao prvo, potrebno je odrediti fizičku veličinu strane na kojoj se štampa, a zatim i površinu koja će biti popunjena tekstom. Ova površina može biti organizovana u jednoj ali i više kolona od kojih svaka ima svoju širinu i razmak u odnosu na sledeću. Slično je i sa ostalim površinama na strani — prostorom za zaglavlja i dno strane na kojima se obično nalazi konstantan tekst i paginacija. Nove veličine se pojavljuju kada se fusnote smeštaju na dno strane — površina koju zauzimaju, način odvajanja od teksta itd. Slično je i sa grafičkim elementima — linijama, krugovima, rasterskim površinama itd. U svim ovim nabrajanjima dominantna reč je površina. Nabrajanje se odnosilo na stranicu knjige koja sa stanovišta rasporeda predstavlja možda najjednostavniji slučaj. Dovoljno je baciti pogled na stranu nekih dnevnih listova da bi se shvatilo koliko raspored površina može biti komplikovan.

Ram do rama

Dakle, kada tekst uđe u *Venturu* njegov sadržaj uglavnom gubi značaj i počinje da se tretira isključivo kao površina koju je potrebno rasporediti na stranu. Zato je i osnovni element kojim program operiše ram (eng. FRAME-frejm). Ramom se određuje prostor koji će biti ispunjen nekim sadržajem (tekstom, slikom). Iako je ovo najčešći slučaj, njima se vrlo često blokiraju i prazne površine na strani. Promena veličine rama i njegovog položaja su glavni elementi kojima se rukuje pri montaži strane.

Dakle, osnovni princip rada *Venture* se bazira na izvršavanju elementarne funkcije koja bi se mogla formulisati na sledeći način: redove teksta čiji je način ispisa definisan karakteristikama tagova treba rasporediti tako da popune površine koje su zadate ramovima.

Ovako uprošćeno postavljen problem deluje kao lak zadatak čije rešavanje ne bi trebalo da predstavlja neki poseban problem. Ipak situacija nije tako jednostavna kao što se čini.

Prvo treba ukazati da se pred *Venturu* postavlja jedan problem koji inače u klasičnoj tehnologiji pripreme — fotoslog — ručna montaža — ne postoji. *Ventura* sama, kao i obični tekst procesori na koje smo navikli, donosi odluku gde treba izvršiti prelazak na novu stranu. Situacija nije jednostavna — princip prostog slaganja teksta do dna strane i zatim prelazak na novu u praksi često traži izuzetke. Nije ni malo prijatno kada neki podnaslov padne na samo dno strane, a tekst na koji se odnosi pređe na sledeću.



SLIKA 1

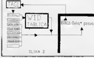
Slično je i sa pasusima čiji samo jedan i to poslednji red prelazi na novu stranu (u žargonu struke ovakav red se naziva „siroče", englezi koriste isti termin — orphan). Iako za ovakve situacije postoji određena vrsta automatike, posledica je često „kraća" strana koja se u ručnoj montaži rešava proređivanjem, ali se u *Venturi* teže koriguje jer iskače iz sistematičnog slaganja, o čemu je već bilo reči u prošlim nastavcima.

Ipak, prednosti automatike su dovoljno velike. Promena nekog od elemenata formata u trenutku daje potpuno novi izgled teksta. Ostaje samo da se proveri da li smo imali dovoljno sreće da se neka od navedenih situacija ne pojavi. Pri ručnom montiranju u navedenom slučaju CEO posao se izvodi od početka. Ne treba ni napominjati koliko prednosti nudi automatsko slaganje recimo fusnota. Ubacivanje nekog pasusa koji remeti njihov raspored automatski se koriguje.

Da bi mogla da obavi svoj posao *Ventura* mora da vodi računa o velikom broju podataka. Na slici 1 je shematski prikazan sadržaj jednog poglavlja teksta koji je kompletno završen i spreman za štampanje. U simbolično prikazanim disketama dati su i produžeci naziva datoteka koje *Ventura* koristi za određenu grupu podataka.

Jedan primer

Poglavlje kao maksimalna količina teksta koju je moguće slagati u jednom prolazu ima samo simbolično značenje. Poglavlje može da sadrži i više stvarnih poglavlja knjige, bez obzira da li su organizovani kao jedna ili više datoteka. Osnovni razlog ovako usvojenog naziva leži u činjenici da *Ventura*, iako može da nastavlja više



SLIKA 2

tekstova, principijelno to može da radi samo ako novi tekst počinje na novoj strani, što je obično situacija sa poglavljima u knjizi. Stvarnu granicu za količinu teksta je prilično teško definisati jer zavisi od mnogo elemenata, o čemu će biti reči nešto kasnije.

Da bi donekle pojednostavili stvar, uzećemo primer jednog poglavlja koje se sastoji iz dva teksta u datotekama 011.TXT i 012.TXT. Pretpostavićemo da su tekstovi kreirani na ranije opisan način, odnosno uz pomoć nekog editora koji generiše klasične ASCII datoteke i da su zatim konvertovani u konačni oblik koji raspoznaje *Ventura* kako je to objašnjeno u prošlom broju. Kao što se vidi na dijagramu, tekstovi predstavljaju samo mali deo onoga što je potrebno da bi se poglavlje kompletiralo.

Unutar teksta su upisani i tagovi, odnosno samo njihovi nazivi. Stvarne karakteristike tagova (koje pismo, koja veličina, koji prored itd.) se zadaju u programu i čuvaju se u STIL datoteci (produžetak naziva .STY). Dakle u STY datoteci su pobrajani svi tagovi koji se koriste zajedno sa karakteristikama svakog od njih. Kada *Ventura* u tekstu naiđe na naziv nekog taga, u STILU pronalazi isti naziv i na osnovu zapisanih karakteristika vrši slaganje pasusa. Da bi slaganje moglo biti obavljeno kako treba, *Ventura* mora da zna tačne karakteristike pisma kojim se slaže, što je zapisano u novoj datoteci, takozvanoj WIDTH tablici (eng. WIDTH=širine), čiji je naziv zapisan u STY datoteci. Ovaj mehanizam je šematski prikazan na slici 2.

Očigledno je da isti STIL može biti primenjen na više tekstova. Dovoljno je da se u tekstu pojavi naziv taga da bi pasus bio automatski složen prema njegovim karakteristikama. Ono što nedostaje je globalni format strane (A4, B5 itd.) pa je zbog toga i on uključen u STY datoteku. Na shemi 1 se uočava i veza između ovog formata i jednog posebnog rama.

U jednostavnom slučaju slaganja knjige sasvim je logično da se odredi jedna površina koja će stalno biti popunjavana tekstom. Da ne bi na svakoj strani stalno ponavljali ram istih karakteristika, Ventura ga uvek podrazumeva. Naziva se „osnovni ram" i tekst se automatski smešta u njega ako se drugačije ne naglasi. Na ovaj način se jednostavniji prelom u mnogome ubrzava. Njegove karakteristike (veličina, položaj na fizičkoj strani, margine itd.) su inicijalno zapisane u STIL datoteci, čime ona dobija kompletne karakteristike potrebne da se definiše osnovni prelom.

STIL bi sam po sebi bio dovoljan kada bi tekst tekao u kontinuitetu, bez ikakvih ilustracija, fusnota, dodatnih tekstova. Ovakva situacija je u praksi veoma retka. Verovatno će biti potreban bar još jedan ram koji će blokirati neku površinu za ilustraciju i slično. Ramova pored osnovnog u praksi obično ima više, tako da su autori *Venture* izabrali da se njihove karakteristike čuvaju u posebnoj datoteci sa produžetkom naziva .CHP. Pošto je ram obično popunjen nekim sadržajem, i ova informacija mora biti zapisana. Zato CHAPTER datoteka na neki način

postaje glavna datoteka koja opisuje sadržaj poglavlja. U našem primeru poglavlja od dva teksta pogledajte kako izgleda sadržaj ekrana na kome su oni ukomponovani (slika 3), a zatim i sadržaj CHP datoteke koja opisuje ovu stranu (slika 4).

Interni format CHP datoteke u principu nema značaja za onoga koji slaže, ali je ovde naveden jer dobro ilustruje uzajamne odnose elemenata od kojih se strana komponuje. Precizno poznavanje rasporeda podataka po datotekama je od presudne važnosti za korišćenje programa. U praksi veoma često nastaju zabune usled nepoznavanja ovih elemenata. Primer: Nakon preloma jedne knjige u kojoj je kreiran i korišćen neki stil, želite da isti stil primenite i na sledeću knjigu. U prvoj ste koristili brojanje strana u centru dna. Kada upišete novi tekst i stari stil, ustanovićete da je brojanje strana isključeno! Razlog deluje misteriozno, ali samo na prvi pogled. Tekst zaglavlja i dna strane je vezan za poseban ram, a kako se ramovi pamte u CHP datoteci a ne u STY, neće biti preneti u novo poglavlje. Isto važi i za ram koji se popunjava fusnotama.



SLIKA 3

Četiri režima rada

Do sada smo govorili o dva različita tipa informacija koje *Ventura* koristi — tagovima i ramovima. Pošto *Ventura* može da radi i sa osnovnim grafičkim elementima (linije, rasteri itd.) a omogućuje i skroman oblik editovanja teksta, uvedena su četiri različita režima rada. Oni su predstavljeni sa 4 ikona na levoj strani ekrana i imaju (redom) sledeća značenja:

1) FRAME režim dozvoljava postavljanje, brisanje i kopiranje ramova, zadavanje njihovih karakteristika i raspoređivanja tekstova i ilustracija unutar njih.

2) TAG režim omogućuje dodeljivanje tagova pasusima teksta, zadavanje njihovih karakteristika, izbacivanje i dodavanje novih tagova itd.

3) EDIT režim dozvoljava unošenje izmena u samom tekstu, izbacivanje i kopiranje njegovih delova, lokalne promene ispisa (eksponent, indeks, lokalna promena pisma itd.) i

4) GRAFIČKI režim omogućuje dodavanje linija, pravougaonika, kružnica i raster površina, njihovo premeštanje, kopiranje, brisanje itd.

U svakom od ovih režima linijski meni na vrhu ekrana dozvoljava pristup različitim opcijama koje odgovaraju izabranom režimu. Ujedno i grafički kursor menja svoj oblik.

Procedura kompletiranja jednog poglavlja se obično izvodi na sledeći način:

1) Na disku se nalazi tekst koji je pripremljen tekst procesorom.

2) Na disku se nalazi nekad ranije korišćeni stil (ili bar DEFAULT.STY koji je „prazan" — nema definisanih tagova osim „BODY TEXT").

3) U FILE meniju bira se opcija LOAD DIFF STYLE i upisuje STY datoteka.

3) Bira se FRAME režim i mišem klikne bilo gde na strani da bi izabrali osnovni ram.

4) U FILE meniju se koristi opcija LOAD TEXT/PICTURE.

5) Zadaje se tip teksi datoteke (ako je korišćena naša preporuka za kreiranje teksta, onda je to ASCII tip).

6) Kada je tekst upisan, automatski se raspoređuje na strane.

7) U FILE meniju se bira opcija SAVE AS NEW STYLE i stil snime na disk pod novim nazivom.

8) Konačno se u FILE meniju bira SAVE AS opcija i sva zajedno snima kao CHP datoteka sa proizvoljnim nazivom.

Nakon ovoga se slobodno mogu menjati karakteristike tagova, dodavati novi tagovi, ramovi itd., odnosno može se započeti rad na prelomu. U pojedinim fazama rada dovoljno je koristiti (iz FILE menija) opciju SAVE (ili skraćeni postupak tasterima Ctrl+S) i sva unete izmene u stilu i ramovima će automatski biti zapisane. Prekid rada i kasniji nastavak je daleko lakši, jer zahteva samo opciju (open FILE meni) OPEN CHAPTER koja vodi računa da se svi potrebni elementi koji čine poglavlje upišu.

Navedena procedura ne deluje baš previše jednostavno, ali se može prihvatiti prilično šablonski (uvek je takva) i garantuje da otvaranje novog poglavlja neće poremetiti raniji rad. Konkretno, ako se za novo poglavlje iskoristi stari stil a ne izvede opcija pod brojem 7, sva izmena koja se u novom poglavlju napravi na tagovima biće prenesena u staro poglavlje u kome je korišćen isti stil, čime će ono praktično biti neupotrebljivo.

Pet iz crtanja

Na slici 1 je drugi element poglavlja CAPTION datoteka koja se kreira automatski bez intervencije korisnika i dobija isti naziv kao i CHP datoteka. Ova datoteka ne postoji uvek, ali je potrebno poznavati njen smisao. Ako se u poglavlju otvori prazan ram i uz pomoć EDIT režima u njega direktno nešto upiše, onda se *Ventura* ▶

GEM

Iako je Digital Research GEM već duže vremena na tržištu, kod nas mu (sasvim nezasluženo) nije posvećivana pažnja. Bilo je čak i nekoliko dezinformacija, počev od toga da naziv programa potiče od engleske reči GEM (izgovara se kao džem — dragulj), do toga da se radi o nekakvom multitasking operativnom sistemu.

GEM — Graphics Environment Manager (što bi se moglo uslovno prevesti kao „upravljač grafičkim okruženjem") je najbolje predstaviti kao grafički dodatak klasičnim operativnim sistemima. Na najnižem nivou GEM predstavlja zbir rutina koje omogućuju crtanje raznih geometrijskih oblika, ispisivanje teksta na „grafički" način (proporcionalno u više različitih pisama — i to ne samo na ekranu već i na štampaču koji podržava grafički režim rada) i rad sa mišem i sličnim digitalizatorima. Na ove rutine nadograđuje se logička organizacija koja omogućuje da se ekran tretira kao složena periferna jedinica na kojoj se mogu otvarati datoteke (prozori) itd. Ugrađene su i neke kompleksnije rutine, kombinovanjem elementarnih, koje pojednostavljuju rad u čestim situacijama kao što je ispisivanje upozorenja, poruka o greškama, izbor datoteka sa diska na osnovu kompletne liste svih koje se nalaze u direktoriju itd. Za kraj je ostavljena posebna verzija upravljačkog programa (DESKTOP.APP, ekvivalent DOS-ovog COMMAND.COM) koja omogućuje interakciju sa korisnikom putem grafičkih simbola.

Rad sa diskom i memorijom nema veze sa GEM-om već je i dalje pod kontrolom DOS-a.

GEM se interno sastoji iz dve glavne celine: VDI modula (Virtual Device Interface) i AES modula (Application Environment Services) koji su dalje izdeljeni na više podcelina. Prvog čini GDOS (Graphics Device Operating System) koji sadrži elementarne grafičke rutine i niz drajvera za direktan rad sa različitim grafičkim jedinicama (video adapteri, štampači itd). Drugi modul (AES) u sebi sadrži srce logičke organizacije čitavog GEM-a (Kernel), zatim Screen Manager i zbir pomenutih rutina višeg nivoa.

Bez obzira na ovako relativno komplikovanu organizaciju, pisanje aplikacija koje će raditi po GEM-om je veoma zahvalan posao iz dva razloga: (a) programer je prisiljen (na svoju korist) da dobro struktuira svoje programe i naročito obrati pažnju na strukture podataka, jer se bez maksimalne preciznosti ne može napisati korektan program i (b) aplikacije su potpuno hardverski nezavisne i instaliranjem odgovarajućih drajvera mogu bez ikakve izmene da rade na velikom broju različito opremljenih računara.

Pisanje aplikacije se može obaviti u bilo kom programskom jeziku (gotovo isključivo se u praksi koristi C), ali je neophodno posedovati i posebnu biblioteku funkcija koja omogućuje poziv i prenos parametara u GEM (takozvane „bind" rutine). Digital Research prodaje paket „GEM Programmers Toolkit" koji sadrži sva neophodna module i alate za lak razvoj aplikacija, ali je, na žalost, njegova cena preko 1000DM.

***Ventura Publisher** odlično ilustruje koliko GEM aplikacije mogu biti kompleksne. Najveći broj korisnika ne zna da karakteristike programa koje ih oduševljavaju (odlična grafika, pregledni meniji, jasne maske za unos podataka itd.) u potpunosti zasluga GEM-a.*

*Zabune oko toga da li GEM omogućuje multitasking ili ne uglavnom potiču od poređenje sa Microsoft-ovim naizgled sličnim programom **Windows**. Iako GEM ima neke sitne elemente višeprogramskog režima rada (deo obnavljenje sadržaja ekrana se može izvoditi u toku rada aplikacije kao nezavisan posao) ne može se porediti za **Windows**-om koji je opremljen mogućnošću da izvodi više programa istovremeno.*

***Ventura** od verzije 1.1 se više ne instalira direktno kao nezavisna aplikacija GEM-a, već je GEM na neki način „zatvoren" za rad sa ovom aplikacijom. Na ovaj način je Xerox izbegao potrebu da prati izmene GEM-a koje diktira Digital Research.*

```
#D 006A 0221 0002 0001 01 01 00 0000 0000 E:\GAL\01.STY
#T 01 E:\GAL\012.TXT
#T 01 E:\GAL\011.TXT
#FN 0000 0000 0000 0000 0000
#FT ** ** ** ** ** ** ** ** **
#P 0001 0000 00 0000
#BT 01 00 00 00 01 0000 0000 3390 27D8 00 0008 0000 0000 0000 0000 0000 **
__ 0589 0589 17FE 0589 0000 0000 0000 0000 1609 0000
__ 0589 0589 0589 17FE 0000 0000 0000 0000 1608 0000
#BT 00 00 01 00 01 0FED 105C 0BD0 1216 04 0000 0000 0000 0000 0000 0000 **
__ 00EC 00EC 00EC 00EC 0000 0000 0000 0000 09FC 0000
__ 00EC 00EC 00EC 00EC 0000 0000 0000 0000 09FC 0000
__ 00 01 08 03 005E 0005 0000 0000 0000 0000 0000
__ 00 00 00 00 0000 0000 0000 0000 0000 0000 0000
```

nalazi pred problemom gde sačuvati ovaj tekst. Pošto ne pripada ni jednom tekstu kreiranom u tekst procesoru, otvara se CAP datoteka i tekst zapisuje u nju. Slična situacija nastaje i sa nekim drugim kraćim tekstovima koji se smeštaju u ramove. U CAP datoteku će biti premešten i deo glavnog teksta ako ga EDIT opcijama „isečete" iz glavnog teksta i prebacite u neki poseban ram.

Situacija sa crtanjem linija i drugih geometrijskih oblika u *Venturi* može da bude posebno zabavna ako se ne poznaje mehanizam po kome se ono odvija. Recimo, odlučite da jedan naslov na prvoj strani uokvirite ramom sa zaobljenim uglovima. Pređete u GRAPHIC režim, nacrtate okvir i zadovoljni njegovim izgledom krenete da obrađujete sledeću stranu teksta. Na njoj vas čeka malo iznenađenje — isti okvir sa prethodne strane! Ovaj put će jeno uokvirivati neki sasvim drugi deo teksta. Razlog leži u činjenici da je i grafika, odnosno njen položaj na strani kao i tekst, uvek vezan za neki ram. Ako se veže za osnovni ram, pojavljivaće se na svakom mestu na kome se i on pojavljuje, dakle na svakoj strani. Za koji ram će grafika biti vezana odlučuje se na osnovu toga koji je poslednji ram proglašen aktivnim u FRAME režimu, što se u praksi lako zaboravlja pa nastanu pomenute situacije.

Sami grafički elementi se pamte u posebnoj datoteci koja ima produžetak naziva VGR a za poglavlje se vezuju, kao i CAP, istim nazivom datoteke.

Ostale opcije

Jedna od najjačih strana *Venture* nije vezana za samo slaganje. Pošto jedno poglavlje može da sadrži ograničeno dugačak tekst, *Ventura* obezbeđuje mehanizam po kome se više poglavlja povezuje u jednu publikaciju. Dovoljno je nabrojati redosledom kojim treba da teku i automatski se otvaraju nove mogućnosti:

1) Automatsko generisanje sadržaja do osam nivoa važnosti. *Ventura* je u stanju da iz teksta povadi pasuse koji imaju određeni tag i uz njih da doda informaciju o broju strane na kojoj se pasus nalazi, tako da je pravljenje sadržaja izuzetno jednostavno.

2) Automatsko arhiviranje na diskete ili druge poddirektorije svih elemenata koja čine publikaciju. Iako se ova operacija svodi na DOS COPY naredbu, njen značaj je mnogo veći jer je to jedini način da se publikacija ispravno premesti iz poddirektorija u kome je kreirana. Pošto je mnogo datoteka „u igri" a nazivi mnogih od njih su zapisani u drugim, sve ove informacije je potrebno izmeniti. Tako prva linija u CHP datoteci koji nosi naziv STIL datoteke mora biti izmenjena ako se publikacija premešta na disk A, jer njen pun naziv nije više, na primer, C:\TYPESET\KNJIGA.STY već A:\KNJIGA.STY.

Pored ove dve najčešće korišćene opcije, obezbeđeno je i automatsko kreiranje referentnih tabela pojmova, štampanje čitave publikacije bez intervencije operatera itd.

Po svemu izuzetno

Ventura po svemu predstavlja izuzetan program — pod uslovom da se tretira na način na koma smo insistirali (maksimalna prethodna priprema). Jednostavnost rada se direktno može meriti odnosom prema tekstu — što manje imamo potrebe da zavirujemo u redove i čitamo šta piše i što više strane posmatramo isključivo kao površine — to vrednost programa dolazi više do izražaja. Možda je ipak najveća vrednost vezana za najmanje vidljive karakteristike — doslednost i jednostavnost u načinu pamćenja informacija vezanih za publikaciju. Što otvara mogućnost spoljnih intervencija koje mogu u mnogome da ubrzaju rad, mnoštvo pomoćnih i veoma korisnih opcija o kojima je bilo reči itd.

Ventura (na žalost) ima i nekoliko ozbiljnih grešaka koji (na sreću) uglavnom mogu biti izbegnute. Ne računajući verziju 1.0, koja je imala ozbiljnije probleme oko dodavanja novih fontova, verzija 1.1 ima velikih problema pri štampanju strana koje imaju linije iz grafičkog režima, ako se za globalni format strane izabere A4 (u najboljem slučaju list je blago išaran grupama tačaka, u najgorem sistem pada). Ovaj nedostatak se lako izbegava upotrebom drugih formata strane i korigovanjem margina da bi se dobio A4 format. Daleko teže situacije nastaju kod glomaznih fusnota — ako su veće od strane, jednostavno nestaju povlačeći sa sobom i delove teksta! Ova situacije je ipak granični slučaj, ali problema ima i kod kraćih fusnota — obično se mnogo veći deo teksta prenosi na novu stranu nego što je to potrebno, ostavljajući prethodnu stranu suviše praznu. I ovo se ponekad može korigovati, doduše na blago čudan način — DODAVANJEM nekoliko redova na strani koji je slabo popunjena!

Program, kad i najveći broj drugih, čak i manje složenih, počinje malo „neartikulisano" da se ponaša kada se dovede na granicu mogućnosti. Tekst duži od dvesta kilobajta ili jako veliki broj ramova lako dovode program na granicu „haosa".

Sve ove uočene slabosti ipak nisu previše značajne. Dovoljno je poznavati navedene ograničenja — tekstovi u jednom poglavlju ne treba da budu duži od stotinjak kilobajta (optimum je oko 50—60 K), broj ramova treba držati u razumnim granicama (100—200 po poglavlju), ne preterivati sa brojem tabulatora na jednoj strani — i ceo posao će teći na izuzetno zadovoljavajući način.

Na kraju da iznesemo i jedan lični utisak. Iako se rad sa *Venturom* predstavlja kao izuzetno jednostavan i lak, smatramo da je to daleko od istine. Suviše detalja traži pažnju i maksimalnu koncentrisanost, pa je potrebno dosta vremena da posao pređe u rutinu, bez koje nema ozbiljnog rada. Tome je najmanje kriva *Ventura* — posao slaganja teksta je takav!

U sledećem, poslednjem, nastavku ove serije pozabavićemo se hardverom — nekim dilemama oko izbora i prikazom karakteristika najčešće korišćene opreme.

Zoran Životić

Knjiga o knjigama

Iako računarstvo često nazivamo informatikom, poplavu informacija koju je kompjuterski bum doneo do sada nisu pratile informacije o informacijama — ne samo što početnik nikada ne zna gde može da pronađe podatke koji su mu neophodni, već i iskusni stručnjaci često satima preturaju po gomilama knjiga i časopisa da bi pronašli tekst, program ili tabelu koja je „sigurno tu negde".

Infogen-ov (*) „VODIČ kroz jugoslovensku računarsku literaturu" pretenduje da reši ovakve i slične probleme. Radi se o knjizi formata 16×23 cm sa plastificiranim koricama na čijih je 260 strana predstavljen abstrakt baze podataka koju je godinama formirao Klub programera Elektrotehničkog fakulteta u Beogradu.

Časopisi su predstavljeni preko registra autora, tema i računara. Svaki tekst objavljen u nekom od devet računarskih ili delimično računarskih časopisa („Računari", „Moj mikro", „Svet Kompjutera", „Tiend", „Mala Računala", „Pilot video", „Galaksija", „Sam", „Radio Amater") predstavljen je podacima o naslovu, autoru, temama, računarima, obimu i dodacima (listinzi programa na bejziku, asembleru, mašinskom jeziku...). Register tema obezbeđuje pronalaženje svih tekstova koji se bave oblašću računarstva koja vas posebno interesuje, registar računara pronalaženje svih tekstova koji su (potpuno ili delimično) posvećeni vašem kompjuteru, a registar autora lociranje svih tekstova vašeg omiljenog publiciste.

U okviru registra knjiga pronaćićete podatke o 700 računarskih knjiga i priručnika koji su štampani u Jugoslaviji između 1981. i 1987. godine — pored dela domaćih autora obuhvaćeni su prevodi, pa čak i u poslednje vreme sve popularniji reprinti stranih knjiga. Svaka knjiga predstavljena je podacima o naslovu, autorima, tipu, izdavaču, godini poslednjeg izdanja i broju objavljenih izdanja. VODIČ opisuje korišćenje usluga Jugoslovenskog bibliografskog zavoda, Narodne i drugih biblioteka, međubibliotečke razmene.

Igračima je namenjen specijalni katalog 2600 igara — u VODIČU su popisane sve igre o kojima je uopšte pisano u domaćoj kompjuterskoj štampi, pri čemu su napisi podeljeni na uvode, rešenja, mape i POKE-ova. Ukoliko vas, dakle, muči neka igra, potražite njeno ime u abecednom spisku i saznajte gde je objavljeno rešenje ili POKE.

Pretplata na „VODIČ kroz jugoslovensku računarsku literaturu" je, uz sve klasične, zasnovana na jednoj novoj i zanimljivoj ideji koja omogućava nabavku knjige bez izlaska iz kuće: pozovete Infogenov dežurni telefon (oglas je negde u ovim „Računarima"), izdiktirate vaše ime, adresu i broj telefona i očekujete poštara. Poštar vam, naravno, neće doneti knjigu (radi se o pretplati) već pozdravno pismo sa otkupninom. Na licu mesta plaćete 13.500 dinara, dobijate potvrdu o uplati i za tridesetak dana očekujete „VODIČ kroz jugoslovensku računarsku publicistiku". Ukoliko ovaj Infogenov metod da zadovoljavajuće rezultate, ne sumnjamo da će mnoge knjige malih izdavača uskoro biti prodavane na slična načine.

(*) INFOGEN je nova izdavačka grupa koja deluje u saradnji i uz podršku Kluba programera Elektrotehničkog fakulteta u Beogradu, Beogradske istraživačke stanice i Mladih istraživača

Dejanove pitalice

Proširenje smederevske tvrđave

Ala je bilo teško! Moramo priznati da nismo pretpostavljali da će 29. pitalica biti tako tvrd orah — jedna magična figura više. Pokazalo se, međutim, da popunjavanje magičnog šestougla nije nimalo jednostavan problem i da je pomoć koju pri tome računar može da pruži sasvim ograničena. Primili smo, sve u svemu, 54 pisma, među kojima smo pronašli svega 25 tačnih i vrlo malo dobro obrazloženih odgovora!

Podsetimo se najpre problema koji smo preuzeli iz tri godine starog Matematičkog lista (zadatak je tada predložio *Jovica Ristić* iz Kruševa). Priča se da je smederevsku tvrđavu (građena tokom 1430. godine) uzidavan kamen nastao raskopavanjem nekog prastarog rimskog naselja. Tom prilikom pronađen je kamen u koji je uklesan pravilan šestougao izdeljen na 24 jednakostranična trougla, u svaki trougao bio je upisan po jedan od brojeva 1—24, tako da zbir u svakom redu koji je paralelan sa glavnom dijagonalom iznosi 75. Učeni Smederevci toga doba zaključili su da je šestougao veoma značajno delo i pripisali mu određene magične osobine — s obzirom da je smederevska tvrđava građena u obliku trougla, pretpostavljeno je da ona predstavlja samo jedno polje magičnog šestougla i da će svi važni događaji na neki način biti povezani sa upisanim brojevima. Zato je odlučeno da se kamen ugradi na neko vidno mesto. Ali, želje su jedno a život drugo: neoprezni radnici su ispustili kamen i on je završio na dnu Dunava!



Naša pitalica je na neki način proširila smederevski šestougao: trebalo je popuniti magičnu šestougaonu strukturu koja se sastoji od 96 jednakostraničnih trouglova tako da dâ zbir po svim dijagonalama paralelnim sa nekom od glavnih dijagonala bude jednak.

Prvi i najznačajniji korak je određivanje zbira svake od dijagonala. Pošto je zbir svih brojeva upisanih u tabelu 96·97/2=4656 a svaki element učestvuje u osam zbirova, zbir po svakoj dijagonali je 4656/8=582. Sledi raspoređivanje brojeva: najbolje je slagati ih po „prstenovima" kojih ima četiri (spoljni prsten se sastoji od 6 ivica odnosno 42 trougla a unutrašnji od šest centralnih polja). Najveći problem predstavlja uklapanje trećeg prstena gde do izražaja može doći kompjutar — zavisno od početno usvojenog (kontrolisanog) RND rasporeda dobija se jedno od brojnih suštinski različitih rešenja. Slika prikazuje jednu od popularnijih varijanti.

Većina korektnih rešenja sastojala se od iscrtanog šestougaonika, kupona i napomene da je do odgovora došeno za dvadesetak sati metodom „probe i greške". Dva pisma su, međutim, bila solidno obrazložena i propraćena programima koje zbog nedostatka prostora ovde ne možemo objaviti. *Leo Bosnić* iz Zadra koji je pronašao deset suštinski različitih rešenja zaslužuje prvu (60.000 dinara) a *Ratko Dinić* iz Tesliča koji je poslao najbrži program pisan za „ZX spektrum" drugu nagradu od 40.000 dinara. Treću nagradu smo izvukli iz grupe tačnih odgovora: 20.000 dinara pripada *Aleksandru Mićiću* iz Banjaluke. Čestitke za interesantna rešenja zaslužuju *Dragan* i *Milan Grbić, Predrag Knežević, Jadran Krašovec* i *Robert Lovrić*.

Zanimljiv dodatak uz ovu pitalicu bio bi problem prebrojavanja suštinski različitih rešenja — verovatno bi se pokazalo da njihov broj uopšte nije mali!

Q zadatak

Iz prošlih „Računara" je, usled nedostatka prostora, „ispalo" O pitalica i rešenje prethodnog zadatka. Evo, dakle, rešenja i duple doze Q zadataka.

RQ5: Trebalo je anagramirati ČAJ NERED tako da se dobije jedna reč. Nije teško: JEDNA REČ. Čitaoci koji su ozbiljno shvatili problem pronašli su i reč DREČANJE i, pod pretpostavkom da postoje, NADREČJE i ANDREJČE.

Q6: Dodavanje jedne neprekidne i neizlomljene prave linije učiniće jednakost: 21 50=10 10 10 korektnom. Nije dopušteno ni na koji način uticati na znak jednakosti (na primer pretvoriti ga u simbol „različito" ili „manje ili jednako"). Svi brojevi zadati su u dekadnom sistemu.

Q7: Uoči poslednjeg kola odbojkaškog prvenstva ekipe „Mladost" i „Vojvodina" imale su jednak broj osvojenih bodova. „Vojvodina" je međutim bila u prilici da sama osvoji titulu prvaka dok je „Mladost" tu titulu u najboljem slučaju mogla podeliti. Kako je to moguće? Napominjemo da se odbojkaška utakmica ne može završiti nerešeno i da zaostalih susreta i Šajberovskih kazni nije bilo. (Ovaj interesantan Q zadatak predložio je Milan Burazov iz Niša.)

Problem susednih cifara

Trideset prva pitalica bavi se među programerima sasvim uobičajenim stepenima dvojke, ali na prilično neuobičajen način. Svi znamo da adresni prostor osmobitnih procesora obično iznosi 65536 odnosno 2↑16 bajta; ovo je ujedno najmanji stepen dvojke koji sadrži dve uzastopne jednake cifre. Slično tome, 2↑24=16777216 sadrži tri, a 2↑41=2199023255552 čak četiri uzastopne jednake cifre. Pronađite najmanji stepen dvojke koji sadrži pet uzastopnih jednakih cifara!

Rešenja, zajedno sa programima koje ste koristili da biste ih dobili, šaljite na adresu: „Računari" (za Dejanove pitalice), Bulevar vojvode Mišića 17, Beograd tako da pristignu pre 25. maja 1988. Sva pisma sa korektnim rešenjima konkurišu za novčane nagrade od 60.000, 40.000 i 20.000 dinara, dok će kuponi (ili njihove fotokopije) na koje je upisan identifikacioni broj učestvovati u godišnjem takmičenju rešavača Pitalica. Identifikacioni broj dobijate tako što u prve tri kućice upišete poslednje tri cifre nečijeg broja telefona, u sledeće dve godinu vašeg rođenja i na kraju dodate dve cifre po izboru. Obratite pažnju da sva rešenja koja šaljete u toku godine nose isti identifikacioni broj.

TV mozgalica

Profesor Percival Prime je veliki ljubitelj prostih brojeva. Priznajući njegove zasluge, pošta mu je dodelila trocifren telefonski broj koji je, prirodno, bio prost. Posle nekoliko godina profesor Prime se veoma uzbudio kada je čuo da je na početak njegovog telefonskog broja iz tehničkih razloga dodata trojka. Srećom, sve je bilo u redu — četvorocifreni broj (3•••) je i dalje bio prost. No, reorganizacija pošte još nije bila završena — na početak četvorocifrenog broja dodata je i jedinica. Tek što se profesor Prime, shvativši da je novi broj (13•••) i dalje prost, oporavio od šoka, pošta mu je javila da će prve dve cifre razmeniti mesta, ali da nema razloga za uzbuđenje — broj (31•••) je i dalje prost. Koji je bio početni broj telefona profesora Prime-a?

Rešenja šaljite na adresu: Televizija Beograd — za „TV Bajt", Takovska 10/3000 Beograd tako da pristignu pre 15. maja 1988.

```
31. Broj 2 ^ ____________ = ________________ ima pet
            uzastopnih jednakih cifara.

Ime i prezime  ____________________________________________

Adresa         ____________________________________________

Mesto          ____________________________________________

Računar        _________________  Vreme  __________________

Identifikacioni broj:  !__! !__! !__!   !__! !__!   !__! !__!
                        iz broja tel.    god.rođ.    po izboru
```

RAČUNARI I INFORMATIKA

Tehnička knjiga

TEHNIČKA KNJIGA JE NAJVEĆI JUGOSLOVENSKI IZDAVAČ KNJIGA IZ OBLASTI RAČUNARA I INFORMATIKE.
Predstavljamo Vam deo ove popularne biblioteke.

1 Adem Jakupović
dBASE III plus
210 strana formata 16 5×23 5 cm
19 000 d

2 Dr Dejan Stajić
INTERFEJSI I MODEMI ZA MIKRORAČUNARE
150 strana formata 16 5 × 23,5 cm
14 500 d

3 Armando Jarne
TURBO PASCAL sa grafičkim aplikacijama
180 strana formata 16 5×23 5 cm
25 000 d

4 Andrew Bennett
MAŠINSKE RUTINE ZA VAŠ COMMODORE 64
Proširenje BASIC-a za C-64 najznačajnijim mašinskim rutinama **128 strana** .. 8 700 d

5 Dejan Ristanović
MAŠINSKO PROGRAMIRANJE NA MIKROPROCESORIMA Z80 I 6502
Ovi osmobitni mikroprocesori ugrađeni su u ZX 81, ZX Spectrum, Spectrum plus Commodore 64 Commodore 128, Amstrad Galaksiju Oric itd Iskoristite u potpunosti mogućnosti Vašeg računara prelaskom sa BASIC-a na mašinski jezik **256 strana** 16.000 d

6 Mr Veselin Petrović i Zoran Mačorinski
COMMODORE 128
C-128 i periferijski uređaji tastatura osnove BASIC jezika komanda za rad sa disketnom jedinicom sistemske naredbe (komanda) grafike spritovi muzika monitor korisni dodaci **190 strana** 13.000 d

7 Bob Steele i Jerry Wallington
RAČUNARI I KOMUNIKACIJE
Knjiga je pisana na osnovu plana i programa za početni kurs informatike u Velikoj Britaniji tokom 1984/85 godina i uspešno se može koristiti za nastavu informatike u VIII razredu osnovne i I i II razredu srednje škola Tekst je izložen postupno i vrlo pregledno a nakon svakog poglavlja dat je test za proveru uspešnosti usvajanja gradiva **224 strane** ... 14.050 d

8 Grupa autora
KUĆNI KOMPJUTERI — Algoritmi i programi za Spectrum i Commodore
Struktura algoritma i BASIC sa specifičnostima Spectrum-a i Commodore-a primene računara u raznim matematičkim disciplinama matematičko modeliranje i igre na računaru **244 strane** . 2.700 d

9 Dragan Majkić
KOMPJUTERSKA GRAFIKA
Osnovno o računarskoj grafici, primena kompjuterske grafike u procesu projektovanja, organizacija crtanja i način unošenja podataka **260 strana** . 16.000 d

10. Clive Gifford
AVANTURE ZA VAŠ ZX SPECTRUM — Listinzi igara
Listinzi najinteresantnijih igara uputstva za igranje osnove za pravljenje sopstvenih igara sličnog tipa **118 strana** ... 1 250 d

11 Dr Mirčeta Danilović
VIDEO-KOMPJUTERSKE IGRE
Podela i vrsta video-kompjuterskih igara sistemi video-kompjuterskih uređaja za njihovu realizaciju programiranje video-igara, ideja i scenario izrada opštag algoritma kodiranje blok-dijagrama u Asembleru 6809 neke zanimljive obrazovne igre **207 strana** . . 2.300 d

12 Dr Dejan Stajić i Dragoslav Jovanović
ODRŽAVANJE I OPRAVKA KUĆNIH RAČUNARA — Spectrum i Commodore
Najčešći uzroci kvarova na ZX Spectrumu detekcija i dijagnoza kvarova zamena neispravnih komponenti hardverski dodaci za ZX Spectrum proširivanje RAM memorije opis i ispitivanje rada mikroprocesora 6510 memorijska mapa računara C-64 o kvarovima memorije i njihovom otklanjanju centroniks interfejs za C-64 **148 strana** 3.350 d

13 Philip Crookal
PROGRAMIRANJE ZA POČETNIKE
Namenjeno svima koji čine prve korake u programiranju Osnovne naredbe BASIC-a, primeri i testovi samostalno programiranje **167 strana** . ... 10.000 d

14 Garry Marshall
AMSTRAD CPC 464 & 664 & 6128 — Primene
Softver za računare AMSTRAD (obrada teksta pomoću programa Amsword baza podataka programi za tabelarne proračune i Easi-Amacalc) Primene bazirane na hardveru (kaseta i diskovi štampači i ploteri) **120 strana** ... . 8 100 d

15 Steve Webb
AMSTRAD CPC 464 — Programiranje u Asembleru
Šta je mašinsko programiranje upisivanje mašinskih naredbi u memoriju nekoliko korisnih mašinskih rutina — pomeranje (scroll) jednog reda teksta ulevo i udesno zvuk lasera zvuk eksplozije bombe itd Dodaci Z80 operacioni kodovi ekranski kodovi program za dizajniranje karaktera o nekim korisnim rutinama iz ROM-a neke nove mašinske instrukcije i rutine **112 strana** . ... 5 000 d

16 John Graham
LIČNI RAČUNARI — Vodič za izbor, korišćenje i primenu
Arhitektura ličnih računara ulazni i izlazni uređaji operativni sistem CP/M razvoj aplikativnih programa primena u poslovne svrhe mreže trendovi u ON LINE aplikacijama izbor sistema **270 strana** .... 3 800 d

17 Grupa autora
NUMERIČKI METODI ZA MIKRORAČUNARE
Objašnjenja i listinzi programa za Commodore i Spectrum iz savremenih numeričkih metoda Interpolacija nelinearne i algebarske jednačine sistemi linearnih i nelinearnih jednačina karakteristične vrednosti i vektori **188 strana** . ... 2.300 d

18 Mr Dragan Pantić
APLIKACIONI PROGRAMI ZA PERSONALNE RAČUNARE — IBM PC/AT/XT I APPLE II C
Tastatura IBM PC i APPLE II C Procesiranje reči pomoću WORDSTAR-a i APPLEWORKS-a Organizacija LOTUS 1—2—3 i njegove osnovne mogućnosti Radna tabela formiranje podataka kopiranje i pomeranje status tabele upravljanje fajlovima poziv za pomoć (HELP) **278 strana** 9.700 d

19 Mr Vojislav Mišić
IBM PC/AT/XT U 25 LEKCIJA
PC hardver DOS — operativni sistem rad sa datotekama obrada teksta jezici linkovanje i biblioteke softver za poslovne ljude korisni dodaci naredbe DOS-a **242 strana** 8 400 d

20 Mr Veselin Petrović i Adem Jakupović
LINIJSKI EDITOR ZA SISTEME EI — HONEYWELL
Knjiga detaljno obrađuje jedan od osnovnih softverskih paketa operativnog sistema računara H6 (ili DPS6) — linijski editor Detaljno su opisana pravila za pisanje direktiva korišćenja specijalnih simbola mogućnosti adresiranja postupci rada kao i sintaksa direktiva uz brojne originalne primere koji ilustruju mogućnosti pomenutog paketa **207 strana** ... . 6 150 d

21 Dejan Ristanović
OBRADA TEKSTA NA RAČUNARU
Obrada teksta na Word Star-u i Word Perfect-u za IBM PC VISA Write AMSWORD za Amstrad 464/6128 itd Upotreba Epson kompatibilnih štampača priprema indeksa pojmova povezivanje računara sa laserskim printerom YU-set slova **210 strana** . ... 14 000 d

22 John Cunliffe
LOGO — Programski jezik
Prvi put na našem jeziku — LOGO za Commodore Atari Spectrum Listinzi programa boja muzika **128 strana** 2 250 d

23 Beško Damjanović
BASIC U NASTAVI MATEMATIKE
Knjiga je namenjena prvenstveno nastavnicima i profesorima osnovnih i srednjih škola kao i učenicima — pre svega zbog svog metodičkog karaktera i velikog broja interesantnih zadataka iz raznih matematičkih disciplina Svaki primer propraćen je tekstualnim objašnjenjem algoritmom i programom **114 strana** 3.400 d

24 Beško Damjanović
ZBIRKA ZADATAKA U BASIC-u (sa rešenjima)
Algoritmi i listinzi programa elementarni zadaci problemski zadaci primene u raznim oblastima **223 strane** 5 600 d

25 Dr Dušan Tošić i dr Vojislav Stojković
PROGRAMSKI JEZIK PASCAL — Zbirka rešenih zadataka iz programiranja
U uvodu je metodološki prikazan proces kreiranja programa na principima strukturnog programiranja Zbirka sadrži veliki broj primera od elementarnih do vrlo složenih **252 strane** .. .. 10.250 d

Upišite znak X preko rednog broja knjige koju poručujete Porudžbinu pošaljite na adresu NIRO TEHNIČKA KNJIGA Beograd 7 jula 26 Isporuke odmah Plaćanje pouzećem
Ove knjige možete nabaviti i u svim većim knjižarama

1 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25

Ime i prezime
Ulica i broj
Broj pošte Mesto

odštampanog grafa da izaberemo fontove za tekst i boje (šrafura) za pojedine serije podataka. Izborom opcije **Quit** ponovo se vraćamo u glavni meni

Ostale komande iz tog menija su

Go otpočinje štampanja prema redoslijedu kojim su izabrani

Pomoću **Align** saopštavamo štampaču da je papir ispravno postavljen na vrh strane.

Page izaziva pomjeranje papira na početak slijedeće strana

Izborom komande **Exit** završavamo rad sa **PrintGraph** programom

Odštampani grafovi su dosta kvalitetni, ali to u velikoj mjeri zavisi i od raspoloživog štampača

Zamjerka bi se mogla staviti na nemogućnost dodatnog uređenja grafova. Ovim programom se istina dobiva jedna korektna čista i standardna forma ali to nije uvijek i dovoljno. Korisniku nije ostavljeno puno mogućnosti da pokaže neku kreativnost. Takvim korisnicima na raspolaganju stoji niz specijalizovanih grafičkih programa. Firma *Lotus* je i sama izbacila jedan takav program pod nazivom *Freelance* (Plus).

Opcije pristupnog menija

Ako smo *Lotus* startali kucanjem imena lotus kao što je predloženo na početku do tabele smo došli preko pristupnog menija. U taj meni se vraćamo i po završetku rada komandom **Quit** iz ovog menija pristupali smo i programu **PrintGraph.** Ostalo je da se objasni namjena preostalih nekoliko opcije

A **View of 1—2—3** je neka vrsta traning programa (**On-Line Tutorial**) i može početnicima dobro doći za upoznavanje za *Lotusom* i njegovim mogućnostima

Install omogućava promjenu podataka o instalaciji programa

Translate je namjenjen konverzaciji podataka u raznim formatima

Međutim nije obavezno da svi ovi programi budu na disku niti ih je neophodno pokretati iz pristupnog menija. Svaki od njih predstavlja nezavisan modul i može se pozvati direktno iz DOS-a. Tako za direktan pristup tabeli treba otkucati **123**, za direktan pristup **PrintGraph**-u **pgraph**, za tutorial **view**, za instalaciju **install**, a za konverziju **trans**.

Nesavršen ali omiljen

Stigli smo i do kraja priče o **Lotusu 1—2—3.** Šta reći kao zaključak? Nadam se da su se iz ovog priloga mogla vidjeti mnoge njegove prednosti. Međutim daleko od toga da nema i mana

Lotus svakako nije najsavršeniji i tehnički najdotjeraniji program koji se može naći na tržištu. Nedostaje mu tekst procesor i komunikacioni dio. Grafičke mogućnosti su relativno skromna. Rekalkulacija je spora i neinteligentna u poređenju sa programima nova generacija (*Quattro Excel*). Raspoloživi set komandi je prilično kompletan i makro mogućnosti su dobre ali to ipak nija programski jezik kakvim raspolažu *Framework* ili *Open Access*. Takođe opšti izgled programa i korisnički interfajs su skromni

Lotus Corporation je i sam pokušao da nadoknadi neka od navedenih nedostataka pa je ubrzo izbacio program pod nazivom *Symphony*. Izgled je bio puno dorađeniji grafičke mogućnosti bolje radilo se za prozorima dodani su tekst procesor i komunikacija ali *Symphony* kao ni bilo koji od navedenih programa nikad nisu dostigli popularnost *Lotusa 1—2—3*. I pored svih nedostataka *Lotus* je naprosto neodoljiv i niko vjerovatno neće pogriješiti ako se opredijeli za rad sa njim. Radi se da takto o industrijskom standardu koji nije uhvatio korijena samo na radnim stolovima već i u srcima korisnika. *Lotus* se može voljeti ili ne ali prema njemu je teško ostati ravnodušan

Tabelarni proračuni

Arif Agović



LOTUS 123

Lotus je proizvod softverske kuće Lotus Corporation i od trenutka svoje pojave na tržištu 1983 godine dospio je u sam vrh svjetskih softverskih bestselera. Svoju popularnost Lotus može da zahvali i činjenici da je bio jedan od prvih integrisanih paketa — programa koji na jednom mjestu omogućuju unos i obradu podataka u elektronskoj tabeli, vođenje manje baze podataka i dobijanje zornih grafičkih prikaza na osnovu podataka iz tabele ili baze. U to vrijeme 16-bitni računari su bili mladi i perspektivni a Lotus je znao da iskoristi njihove prednosti za postizanje maksimalne brzine i efikasnosti. Uz sve to ovaj program je dosta jednostavan za korišćenje. On omogućuje i neprogramerima da bez kompleksa koriste računar u svom svakodnevnom poslu ali ne sputava ni kreativnost radoznalijih i spretnijih korisnika ostavljajući im mogućnost da prave sopstvene aplikacije

Osnovna namena ovog priloga je da široki krug korisnika upozna sa ovim popularnim programom. Osim toga priča o Lotusu je ujedno i priča o jednoj dosta širokoj klasi poslovnog softvera — tzv speadsheet programima odnosno programima za tabelarne proračune. Još od pojave VisiCalc-a programi ovoga tipa ne prestaju da zaluđuju poslovni svijet. Osim Lotusa na tržištu postoji čitav niz sličnih programa. Jedni su specijalizovani samo za tabelarne proračune (Multiplan, Supercalc, Javelin, Boeing Calc) a drugi spadaju u klasu integrisanih paketa i objedinjavaju tabelu bazu podataka procesor teksta grafiku itd. (Symphony Framework Open Access Enable Ability) Princip rada sa tabelom je kod svih gotovo isti. Mnogi su se pri tome svjesno ugledali na Lotus.

O Lotusu su napisani tomovi i tomovi knjiga pa ne treba ni očekivati da se na ovako ograničenom prostoru mogu osvijetliti svi detalji ali neki principi se ipak mogu dati. Ovaj tekst bi trebalo da posluži samo kao podsjetnik ili početni vodič u radu sa ovim programom. Ambicioznijim korisnicima preporučujemo da ako ga nemaju svakako nabave kompletno uputstvo kao i da pročitaju neku od mnogobrojnih knjiga

Možda još treba napomenuti da će se sve odnositi na verziju (release) 2.0, koja pored kompatibilnosti sa starom verzijom ima i čitav niz novih opcija

Elektronska tabela

Lotus spada u grupu tzv "spreadsheet" programa — osnovu njegovog rada čini veliki elektronski list ili elektronska tabela. List je podijeljen na 256 kolona i 8192 redova. U presjeku redova i kolona formiraju se ćelija. Ima ih tačno 2097152

Da bi se stekao utisak kolike su stvarne dimenzije tabele to jest koliki bi bio njen papirni ekvivalent pretpostavimo da su sve ćelija iste veličine dimenzija 0.7×3 centimetara. Jednostavnim množenjem dolazimo do dimenzija 57.344×7.68 metara! Taj papir bi zauzimao površinu od 440.40 kvadratnih metara. Uporedite to sa na primjer površinom prosječnog stana

Pošto smo se uvjerili da nam prostora neće nedostajati da razmotrimo šta se sa takvom tabelom može urediti

Elektronska tabela je idealna za rješavanje svih brojčano orijentisanih problema naročito kada je potrebno brzo proigrati razne strategije variranjem pojedinih parametara. Uz izvjesna uopštavanja *Lotus* se može smatrati i sistemom za podršku odlučivanju (Decision Support System)

Mnoštvo svakodnevnih problema može se veoma uspješno i efikasno rješavati i bez poznavanja programiranja. Uobičajeni postupci se uz to mogu opet bez programiranja automatizirati a uz malo više truda mogu se napraviti i kompletna aplikacije. Ne treba zaboraviti ni mogućnost grafičkog predstavljanja podataka — kažu da je jedna slika vrijednija nego hiljadu brojeva.

Bliski susret s Lotusom

Instalisanje i pokretanje

Lotus dolazi na 6 (šest) disketa
1-2-3 System Disk,
PrintGraph Disk
Utility Disk
Install Library Disk,
A View of 1-2-3 i

Backup System DiskŠesti disk je u stvari samo rezervna kopija prvog

Instalacija programa je dosta jednostavna i o njoj se brine poseban *Install* program koji se nalazi na *Utility* disku. Dovoljno je samo slijediti instrukcije programa.

Da bi se, nakon instaliranja *Lotus* pokrenuo sa diskete ili diska, potrebno je u DOS-u otkucati njegovo ime, tj. LOTUS. Poslije nekoliko trenutaka na ekranu će se pojaviti logo, a zatim i *Lotusov* pristupni meni **(1-2-3 Access System)**, iz koga pristupamo samom programu ili nekim drugim opcijama. Po meniju se krećemo kurzorskim strjelicama, a izbor se vrši tipkom Enter. Za početak rada sa tabelom treba izabrati opciju 1-2-3. O ostalim opcijama biće riječi kasnije.

Ekran i tastatura

Nakon nekoliko sekundi i kopirajt poruke pojaviće se *Lotusov* radni ekran. Ekran je podijeljen na nekoliko oblasti.

Najveći dio zauzima, razumljivo, sama tabela. Ona je ograničena ivicama na kojima se nalaze oznake kolona i redova. Kolone su označene slovima, a redovi brojevima. Pošto nema dovoljno slova da se označi svih 256 kolona, kolone se označavaju i kombinacijom slova. Tako, nakon kolone sa oznakom Z dolaze kolone sa oznakom AA, AB, ... AZ, BA, BB, ... — sve do kolone IV. Oznake redova kreću se u rasponu od 1 do 8192. To je značajno unapređenje u odnosu na verziju (Release) 1 koja je imala samo 2048 redova.

Na ekranu, naravno, nije moguće odjednom vidjeti čitavu tabelu, već samo jedan njen mali dio, tako da možemo smatrati da je ekran u stvari prozor koji možemo da pomjeramo po tabeli i tako promatrati njen sadržaj. Inicijalno, taj prozor pokazuje gornji lijevi ugao — kolone A do H i redova 1 do 20.

Kao što je već rečeno, u presjecima redova i kolona nalaze se ćelije. Svaka ćelija je jednoznačno određena tim koordinatama. Može se primijetiti da je ćelija u gornjem lijevom uglu (koordinate A1) posebno osvijetljena. To je zato jer se na njoj nalazi pokazivač aktivne ćelije **(cell pointer)**. Pomoću tog pokazivača određujemo ćeliju sa kojom ćemo raditi.

Sljedeća oblast, koju čine gornje tri linije iznad ivice sa oznakama kolona, naziva se kontrolni panel.

U prvoj liniji je prikazana koordinata trenutno aktivne ćelije (ćelije na kojoj je pokazivač), format prikaza, status zaštite i sadržaj ćelije.

U istoj liniji, ali u krajnjem desnom uglu nalazi se indikator stanja ili načina rada **(mode indicator)**. Kada na tom mjestu stoji READY, program je spreman da prihvati naše komande. U trenucima kada je zauzet nekim drugim poslom, na primjer, kada radi sa diskom, na tom mjestu se pojavljuje WAIT.

Sljedeća, druga, linija služi za prikaz ili sadržaj ćelije koji se unosi ili mijenja, ili komandnog menija ili poruka kada neka od komandi zahtijeva dodatne parametre.

Treća linija je rezervisana za dodatno objašnjenje komandi iz menija.

Još je preostala zadnja linija ekrana, koja je rezervisana za prikaz poruka o greškama i statusa tastature.

Nakon opisa ekrana, vratimo se opet tabeli da bismo razmotrili načine kretanja po njoj.

Pokazivač ćelije jednostavno pomjeramo u željenom pravcu koristeći kurzorske strjelice. Ako pokazivačem dođemo do ivice ekrana, prozor kroz koji gledamo tabelu pomjeriće se u tom pravcu (naravno ako u tom pravcu ima slobodnih ćelija). Na praznoj tabeli to pomjeranje neće biti tako očigledno, ali može se primijetiti po promjeni oznaka redova ili kolona na ivicama. Ivice se nikada ne gube sa ekrana.

Po tabeli se možemo kretati i u većim koracima. Tipke **PgUp** i **PgDn** se koriste za pomjeranje za stranicu gore, ili dole, a lijeva i desna kurzorska strjelica u kombinaciji sa **Ctrl** tipkom za pomjeranje za stranicu lijevo, ili desno. Isti efekat se može postići i korištenjem tipki **Tab** (stranica udesno) i **Shift** i **Tab** (stranica ulijevo).

Home tipka vraća pokazivač na ćeliju A1. Uz pomoć tipke **End** i neke od kurzorskih strjelica može se brzo doseći do ivica tabele. Tipke se ne pritišću istovremeno, već se prvo pritisne tipka **End**, a zatim neka od strjelica.

Postoji i mogućnost direktnog pozicioniranja pokazivača na određenu ćeliju navođenjem njenih koordinata. Za to se koristi funkcijska tipka **F5 (GoTo)**.

Tipovi podataka

Prije početka rada treba nešto reći i o tipovima podataka koji se mogu naći na tabeli.

Podaci se mogu pojaviti u tri vida: kao komentari, tj. labele, kao brojevi i kao formule.

Labele su jednostavno nenumerički podaci, komentari i tekst, koji se koriste za opisivanje i objašnjenje ostalih podataka na tabeli. Tabela, u principu, može raditi i bez labela, ali je pitanje da li bi ko znao šta takva tabela radi.

Šta su brojevi ne treba puno objašnjavati. Uglavnom se koriste kao ulazni podaci.

Formule služe za dobijanje rezultata na osnovu ulaznih podataka.

Za svaku vrstu podataka postoje neka pravila.

Broj, osim cifrom, može počinjati znakovima + − . $ ili decimalnom tačkom. Broj može imati samo jednu decimalnu tačku i ne smije sadržati zarez, razmak, ili neko slovo izuzev slova E koji se koristi za oznaku eksponenta u naučnoj notaciji.

Ako podatak počinje slovom, tretira se kao labela. Ako labela treba da počne brojem, ili nekim specijalnim znakom, treba koristiti posebne prefikse da bi se naznačilo da je taj podatak labela, a ne broj. Osim za označavanje labele, prefiksi se koriste i za formatiranje. Apostrof (') ispred labele znači da će labela biti poravnata po lijevoj ivici ćelije, navodnici (") poravnavaju labelu po desnoj ivici, znak (^) se koristi za centriranje labele, a uz pomoć obrnute kose crte (\) prvi znak labele ponavlja se u čitavoj ćeliji. Ovo je posebno zgodno kada treba ispisati crte.

Formule su izrazi sastavljeni od brojeva, ćelijskih referenci (koordinata) i *Lotusovih* funkcija međusobno povezanih matematskim operacijama. Na raspolaganju su sve uobičajene operacije, a njihovi simboli i prioritet su sljedeći:

Preporuke za globalni dizajn

Na kraju još nekoliko riječi o dizajnu tabela. Baš kao što i svaki dobro napisan program ima jasnu strukturu, tako i dobro kreirana tabela mora biti čitka i razumljiva. Struktura tabele treba da prati tok proračuna i da ima neki logički slijed. Dobro dizajnirana tabela obično nije velika i iz nje se na prvi pogled vidi o čemu se radi. Pri kreiranju ne treba štedjeti na labelama. Obično se na tabeli, pogotovu ako se radi o većem proračunu, nalazi nekoliko cjelina. Tu je tabela sa glavnim pregledom i nekoliko satelitskih tabela u kojima se računaju vrijednosti za glavnu. Važni rezultati su jasno izdvojeni.

Tako bi tabela koja prikazuje poslovanje neke radne organizacije sadržala samo konačne podatke o poslovanju svojih osnovnih organizacija. Svaka pojedinačna osnovna organizacija imala bi opet svoju zasebnu tabelu. Rezultati te tabele prenosili bi se direktno u glavnu.

Satelitske tabele mogu biti i zasebne, da bi se sa glavnom tabelom kombinovale komandom **/File Combine.**

Preporučuje se da u posebnom dijelu izdvoje svi parametri podložni promjenama. To znatno olakšava sve vrste analiza. Mada je uobičajeno, nije uvijek neophodno da se konačni rezultati nalaze na dnu tabele. Nekad je upravo zgodnije da budu na početku.

Ako smo napravili tabelu za praćenje stanja tekućeg računa, zgodnije nam je da čim učitamo tabelu na samom vrhu vidimo konačne cifre, već da prelazimo nekoliko ekrana dok ne dođemo do reba sa krajnjim rezultatom.

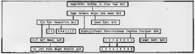
Štampani izlaz

Nakon svih analiza rezultate je obično potrebno dati i u pisanoj formi. Papir je još uvijek nezaobilazan medij u komunikaciji među ljudima. Ipak je još uvijek lakše ponijeti i nekome pokazati par listova papira nego sa sobom vući računar.

U izvještajima tabela se može dopuniti i dodatnim tekstom, objašnjenjima i preporukama. Na žalost *Lotus* raspolaže samo rudimentalnim mogućnostima za rad sa tekstom. Zato je obično potrebno tabele prenijeti u neki program za obradu teksta i tamo ih dodatno urediti. Da bi to postigli, prilikom štampanja izlaz treba preusmjeriti u datoteku, a zatim tu datoteku učitati u tekst procesor. Ako se koristimo *SideKick*-om, možemo direktno se ekrana isjeći tabelu i prenijeti je u *Notepad*. Uz to, više firmi je napravilo dodatne tekst procesore koji rade sa *Lotusom*.

Noviji programi za obradu teksta omogućuju da se, osim tabele, u tekst ubace i slike. No, pozabavimo se generalno mogućnostima štampanja poslovne grafike načinjene *Lotusom*.

Rad sa Print-Graph programom



Da bi se na štampaču ili ploteru dobili grafovi kreirani u toku rada sa Lotusom, potrebno je preći u poseban program za štampanje koji se zove **Print Graph.** Naravno, podrazumijeva se da su prethodno grafovi snimljeni na disk komandom **/Graph Save**. Ta datoteke prepoznaju se po nastavku PIC. Do programa stižemo preko pristupnog menija izborom opcije **Print Graph.**

Prije samog štampanja treba obaviti nekoliko operacija.

Ako prvi put radite sa **PrintGraph-om** neophodno je programu saopštiti osnovne informacije o instalaciji. Zato iz komandnog menija biramo opciju **Settings** i **Hardware**. Ako je to jednom urađeno, postupak nije potrebno ponavljati, osim ako se nije promijenila instalacija. Tekuća instalacija se opcijom **Save** može sačuvati u datoteci PGRAPH.CNF. Opcija **Reset** zamjenjuje trenutne parametre instalacije onima iz datoteke PGRAPH.CNF. Izborom opcije **Quit** vraćamo se u glavni meni.

Nakon specificiranja hardverske instalacije potrebno je izabrati šta ćemo štampati. **Image-Slect** opcija omogućava da se jedan ili više grafova izabere za štampanje. Grafovi se biraju iz ponuđene liste na kojoj se nalaze sve datoteke sa nastavkom PIC iz tekućeg direktorija. Po listi se krećemo kurzorskim strijelicama, a izbor potvrđujemo ili poništavamo razmaknicom (spacebar). U toku izbora označeni graf se može pregledati pritiskom na tipku **F10** (GRAPH). Pritiskom na **Return** završavamo sa biranjem i vraćamo se u meni.

Vjerovatno ćemo htjeti da prije štampanja naš graf još malo doradimo i uljepšamo. Za to je potrebno izabrati opciju **Image** iz **Settings** menija. Ona nam omogućava da odredimo dimenzije i orijentaciju

Kreiranje baze

Kreiranje baze u *Lotusu* je u principu isto kao i kreiranje tabele, možda je čak i jednostavnije. No prije samog kreiranja potrebno je napraviti određenu analizu podataka koji će se pojaviti u bazi. Prvo treba odrediti skup potrebnih podataka a zatim i nazive i maksimalne dužine polja u koja će se ti podaci unositi.

Ako, na primjer, želimo da napravimo bazu podataka softvera, biće nam potrebna polja kao što su: NAZIV (20), PROIZVOĐAČ (15), TIP (10), BROJ DISKETA (2) itd. U zagradama su navedene pretpostavljene širine polja.

Nazivi polje unose se u jednom redu, u susjedne kolone, kao labele. Širina kolone podešava se komandom (**Worksheet Column-Width**. Na ovaj način nastaje zaglavlje baze.

Kada završimo sa kreiranjem strukture baze, možemo početi sa unosom podataka. Podaci se unose na uobičajeni način. Pokazivačem se dođe na ćeliju (polje) i ukuca se odgovarajući podatak. Za promjenu već unesenih podataka možemo se poslužiti tipkom F2 (EDIT). Kartone iz baze možemo izbaciti na isti način kao što brišemo redove iz tabele. Postoji i drugi način, ali će o tome biti više riječi kasnije.

Baza se podacima se u komandama koje rade sa bazom obično označava kao opseg podataka (**Data Range**), ili ulazni opseg (**Input Range**).

Sortiranje baze

Sortiranje baze podataka predstavlja jednostavno svrstavanje kartona po nekom redoslijedu. Redoslijed može biti rastući ili opadajući. Da bi se izvršilo sortiranje potrebno je naznačiti ključno polje, to jest polje po kom će se izvršiti sortiranje. Osim primarnog, može se navesti i pomoćni, sekundarni ključ, koji omogućava sortiranje unutar sortiranje.

Tako za primarni ključ možemo izabrati polje PROIZVOĐAČ, a za sekundarni polje NAZIV. Rezultat je baza u kojoj su programi izdvojeni po proizvođačima, a unutar svake grupe sortirani po abecednom redu. Sortiranu bazu daleko je lakše pregledati nego nesortiranu.

Za sortiranje baze koristimo se komandom **/Data Sort.**

Pretraživanje i izdvajanje

Baza ne bi imala smisla da je nije moguće pretraživati i u njoj pronaći i prikazati kartone koji zadovoljavaju određeni kriterij.

Baza se pretražuje komandom **/Data Query**. Osim ulaznog opsega, za koji je već rečeno šta predstavlja, ovdje se spominju i opseg kriterija **(Criterin Range)** i izlazni opseg **(Output Range)**. Kako se označavaju, već je objašnjeno u dijelu sa komandama, pa ćemo se ovdje više pozabaviti njihovom funkcionalnošću.

Prvo šta treba urediti je kreirati opseg kriterija. Najlakši način za to je da se iskopiraju svi ili samo neki nazivi polja. Potrebna su ona polje po kojim će se vršiti pretraživanja. Kriterij pretraživanje zadajemo tako da ispod odgovarajućeg naziva polja upišemo željeni sadržaj.

Tako, ako želimo da pregledamo sve programe čiji je proizvođač Microsoft, u ćeliju ispod labela PROIZVOĐAČ upisaćemo Microsoft.

Ako sada iz menija izaberemo opciju **Find**, *Lotus* će u bazi osvijetliti prvi karton koji zadovoljava postavljeni uslov. Strijelicama za gore i dole možemo se pomjerati do ostalih kartona. Izborom opcije **Extract**, te kartone možemo izdvojiti u poseban dio tabele, prethodno definisan kao izlazni opseg.

Osim ovog jednostavnog uslova mogu se zadavati i složeniji. Možemo poželjeti, na primjer, da izdvojimo sve programe tipa tekst ili baza čiji su proizvođači Microsoft ili Ashton-Tate. Ovaj kombinovani uslov zadaćemo tako što ćemo ispod labele TIP upisati tekst, a zatim u istu kolonu, u red ispod, baza. Time smo ostvarili uslov tekst ili baza. Uslov Microsoft ili Ashton-Tate ostvarićemo na analogan način upisivanjem odgovarajućeg sadržaja u redove ispod labele PROIZVOĐAČ. Još je samo ove dodatne redova potrebno uključiti u opseg kriterija.

Ako želimo da vidimo koji se sve tipovi programa nalaze u bazi, u red ispod labele TIP u opsegu kriterija nećemo ništa upisati (upisaćemo blanko). Takođe, za izlazni opseg navešćemo samo polje TIP. Izborom Komande **Unique,** u izlaznom opsegu pojaviće se svi različiti tipovi programa u bazi.

Brisanje kartona

Na isti način kao i kod pretraživanja iz baze se mogu trajno ukloniti (obrisati) kartoni koji zadovoljavaju neki uslov. To se radi opcijom **Delete.** Ova komanda ne zahtjeva izlazni opseg, jer se brisanje vrši u ulaznom opsegu. Prilikom brisanja ostali kartoni se pomjeraju tako da popunjavaju nastale praznine.

Ako želimo da obrišemo sve programe čiji je proizvođač IBM, u opsegu kriterija ćemo ispod labele PROIZVOĐAČ upisati IBM, a zatim izabrati opciju **Delete**. Naknadnim pretraživanjem možemo se uvjeriti da su iz baze uklonjeni svi kartoni koji zadovoljavaju navedeni uslov.

( )	zagrada koje se koriste za grupisanje
	stepenovanje
–	negativan broj
*	množenje
/	dijeljenje
+	sabiranje
–	oduzimanje

Kao ni brojevi, ni formule ne dozvoljavaju razmake, niti smiju počinjati nekim znakom alfabeta. Dodatni znakovi kojima može početi formula, u odnosu na već navedene za brojeve su: ((mala zagrada), @ (ludo a ... majmunski znak) i # (heš, taraba). U principu, ako se stekne navika da se svaka formula započinje znakom + ili (, nikada neće biti problema. Rezultat formule je obično broj, ali mogući su i drugačiji izlazi.

Prvi koraci

Kreiranje tabele

Probajmo sada da kreiramo neku manju tabelu. Neka to, na primjer, bude problem praćenja prodaje neke grupe proizvoda.

Prvo treba upisati odgovarajuće labele i na taj način dati tabeli neku formu.

Labelu upisujemo tako što pokazivačem dođemo na željenu ćeliju, neka to bude A1, i otkucamo, na primjer, 'Proizvod. Prefiks ' označava da će labela biti poravnata po lijevoj ivici ćelije. Moglo se i bez tog prefiksa, jer se lijevo poravnavanje labela podrazumijeva. Unos završavamo pritiskom na tipku **Enter**, ili, što je bolji način, na strjelicu u pravcu ćelije u koju ćemo unositi sljedeći podatak.

U ćeliji B1 upišimo "Količina, u C1 "Cijena i u D1 "Iznos. Zbog prefiksa " sve tri labele biće poravnate po desnoj ivici.

Pređimo u ćeliju A2 i otkucajmo \–. U ćeliji će se pojaviti crta sastavljena od znakova –. Ponovimo taj postupak u ćelijama B2, C2 i D2. Tabela je već dobila neku formu.

Uzmimo za početak da pratimo samo pet proizvoda i njihova imena upišimo u ćelije A3 i do A7. U osmom redu formirajmo ponovo crtu, ponavljajući postupak kao u ćelijama A2 do D2. Time je tabela dobila konačnu formu.

Sada je na redu unos brojčanih podataka u kolone označene sa labelama **Količina** i **Iznos.** Pošto se sada unose brojevi, nikakvi prefiksi nisu potrebni.

Nakon unosa brojeva tabela bi trebalo da izgleda ovako:

Proizvod	Količina	Cijena	Iznos
Roba1	103	154 50	
Roba2	61	293 15	
Roba3	47	992 75	
Roba4	23	1045 20	
Roba5	18	1678 60	

Naravno, svako je mogao unijeti drugačije brojeve.

Sada preostaje još da se izračunaju iznosi dobijeni prodajom pojedinačnih proizvoda, kao i sumarni iznos.

Za tu svrhu potrebno je kreirati odgovarajuće formule. Princip je jednostavan. Potrebno je pomnožiti količinu sa jediničnom cijenom i rezultat upisati u ćeliju u preostaloj koloni. Kako su podaci o količini i cijeni već uneseni u odgovarajuće ćelije, pomnožićemo, u stvari, sadržaje ćelija.

Krenimo od ćelije D3. Potrebno je pokazivačem doći na nju i otkucati +B3*C3, tj. našu formulu. Nakon pritiska na **Enter**, ili strjelicu, u ćeliji D3 će se pojaviti rezultat dobijen formulom, a ne samo formula, iako je formula i dalje stvarni sadržaj ćelije.

Postupak bi se mogao ponoviti i za ostala ćelije u koloni Iznos. Formula bi bila ista, samo bi se mijenjala koordinata reda ćelija koje se koriste. Tako bi u ćeliju D4 trebalo upisati formulu +B4*C4, u D5 +B5*C5 itd.

Još treba izračunati sumu kolone Iznos. U ćeliji D9 treba upisati +D3+D4+D5+D6+D7.

Nakon završetka upisa formula tabela treba izgledati ovako:

	A	B	C	D
1	Proizvod	Količina	Cijena	Iznos
2				
3	Roba 1	103	154 50	15913 50
4	Roba 2	61	293 15	17882 15
5	Roba 3	47	992 75	46659 25
6	Roba 4	23	1045 20	24039 60
7	Roba 5	18	1678 60	30218 40
8				
9				1 35E+5

Može se primijetiti da je suma ispisana malo drugačije od ostalih brojeva. Razlog tome je što broj nije mogao stati u ćeliju, pa je ispisan u tzv. naučnom (scientific) formatu. Dok se drugačije ne odredi, program se nalazi u opštem (general) formatu i sam prilagođava način ispisa brojeva.

Za kreiranje ove tabele trebalo vam je izvjesno vrijeme, možda čak i duže nego da ste proračun uradili pomoću džepnog kalkulatora. Kada bi se ostalo samo na tome, elektronska tabela ne bi imala puno smisla. Prave prednosti se uočavaju kada dođe do promjena ulaznih podataka. Promjenom bilo kojeg podatka u kolonama **Količina** i **Cijena**, doći će do automatskog preračunavanje tabele — bez ikakvog pisanja novih formula. To otvara nove mogućnosti i komplikovane analize koje se zasnivaju na variranju ulaznih parametara praktično svodi na igru.

Osim toga, tabela se mogla i brže kreirati — korištenjem **Copy komande** i *Lotusovih* funkcija.

Kao što se i iz ovog primjera moglo vidjeti u tabeli se često javljaju iste formule. U našem slučaju u ćelijama od D3 do D7 nalaze se praktično iste formule. Probajmo da formule upišemo na slijedeći način.

U ćeliju D3 upišimo +B3*C3 (stara formula). Osim direktnog upisivanje koordinata u pisanju formula možemo koristiti i pokazivač ćelija i tzv. metodu „cursor pointing-a". Princip je jednostavan — kada nam u formuli zatreba neka ćelija, umjesto da po tabeli tražimo koordinatu, jednostavno dođemo pokazivačem na tu ćeliju, a koordinata se sama pojavljuje u formuli.

Probajmo tu metodu u praksi. Otkucajmo znak +, a zatim pritisnimo kurzorsku strjelicu za gore (ili dole). U drugoj liniji ekrana, tamo gdje je prikazan sadržaj koji se unosi, pojaviće se koordinata ćelije na kojoj se nalazi pokazivač. Pomjeranjem pokazivača mijenja se i koordinata.

Da bismo napisali našu formulu, dođimo pokazivačem na ćeliju B3, otkucajmo znak *, pritisnimo strjelicu za gore, pokazivačem dođimo na ćeliju C3, pritisnimo Enter i formula je gotova. Jednostavno, zar ne? Naravno, prilikom pisanje formula slobodno možemo mješati obe načina.

Da bismo dovršili tabelu, dok smo na ćeliji D3 pritisnimo tipke / i c. Time smo pozvali komandni meni i izabrali komandu **Copy**. Ta komanda zahtjeva da se navedu dva opsega — prvi šta se kopira i drugi gdje se kopira. Pod opsegom se podrazumijeva pravougaona oblast ćelije, a označava se koordinatama ćelija u gornjem lijevom i donjem desnom uglu. U našem slučaju prvi opseg je samo jedna ćelija pa kucamo D3..D3 (ili samo pritisnemo tipku **Enter** dva puta). Taj sadržaj treba da kopiramo u ćeliji D4 do D7, pa kao drugi opseg kucamo D4..D7. I prilikom određivanja opsega možemo se koristiti pokazivačem. Izabrani opseg biće prikazan inverzno.

Kada se komanda izvrši u ćelijama će se pojaviti isti rezultati kao i prije. Pregledom ćelija može se ustanoviti da su i formule korektne.

Šta se, u stvari, desilo? Program je sam, prilikom kopiranja, prilagodio formule. To je bilo moguće jer su sve koordinate navedene u formulama relativne. Formulu +B3*C3 upisanu u ćeliju D3 bi u stvari trebalo čitati kao: pomnoži sadržaj ćelija koje se nalazi dva mjesta lijevo sa sadržajem ćelije koja je jedno mjesto lijevo od ćelije sa formulom. Ovaj princip omogućeva da se formule lako kopiraju i automatski prilagođavaju poziciji na kojoj se nalaze.

Koordinate mogu biti i apsolutne, za slučaj kada nam je potrebno striktno obraćanje nekoj ćeliji. Da bi se koordinata označila kao apsolutne ispred oznake kolone i/ ili reda treba dodati simbol $. Tako koordinata $A1 apsolutno označava kolonu A, dok je red i dalje relativan, koordinata A$1 apsolutno označava red 1, a kolona je relativna, dok koordinata A1 apsolutno označava ćeliju A1. Prilikom kopiranja ne dolazi do prilagođenja apsolutnih koordinata.

Preostala nam je još formula za računanje sume. Formalno nema ničeg lošeg u formuli +D3+D4+D5+D6+D7, osim što je praktično neupotrebljiva kod većeg broja podataka. Zamislite samo da na ovaj način sumirate nekoliko desetina ili čak stotina podataka. Za takve prilike postoji posebna funkcija — @ **sum**. Sintaksa funkcije je @ **sum** (opseg).

Novu formulu ćemo otkucati na slijedeći način: otkucajmo @ **sum** (, pritisnimo strjelicu za gore, pokazivačem dođimo na prvu ćeliju opsega (D3) i otkucajmo . (tačka). Time smo fiksirali prvu koordinatu. Na isti način označimo i drugu koordinatu (D7) i otkucajmo). Pritiskom na **Enter** završićemo pisanje formule. Daleko jednostavnije i elegantnije nego da smo nabrajali sve sabirke.

U slučaju da su neke formule pogrešno napisane, mogu se popraviti bez potrebe za ponovnim kucanjem. Da bi se pristupilo već unijetoj formuli, koristi se funkcijska tipka **F2**(EDIT).

Rad sa diskom

Na kraju, preostalo nam je još da naučimo kako da sačuvamo našu tabelu na disku (disketi). Za to je potrebno otkucati **/fs (File Save).** Na ekranu, u trećoj liniji, pojaviće se nazivi svih datoteka na radnom disku. Pošto je ovo tek kreirana tabela, program će zahtjevati da se unese neko ime. Ime može biti dugačko maksimalno 8 znakova i ne smije sadržati razmake ili znakove interpunkcije. Nastavak (extension) se ne kuca, a program će automatski dodjeliti nastavak WK1. U verziji 1 nastavak je bio WKS. Već ova razlika ukazuje na nekompatibilnost. Na sreću, tu je program *Trans*, koji će izvršiti konverziju formata. Programu se pristupa iz **1—2—3 Access System-a.**

Ako se tabela želi spremiti na jedinicu ili direktorij koji nije tekući, treba prije imena dodati oznaku jedinice, ili put. Ako na disku nema dovoljno mjesta, program će se oglasiti porukom „disk full". Trebalo bi steći korisnu naviku da se u kraćim vremenskim intervalima tabela na kojoj radimo sprema na disk, kako bi u slučaju nekih nepredviđenih okolnosti bilo izgubljeno što manje rada.

Da bi se tabela pročitala sa diska treba otkucati **/fr (File Retrieve).** U trećoj liniji pojaviće se liste raspoloživih datoteka. Ime željene datoteka može se izabrati iz liste, korištenjem kurzorskih strjelica, a može se i direktno otkucati.

Rad sa programom završavamo ako otkucamo **/q (Quit).** *Lotus* će za svaki slučaj provjeriti vašu namjeru jer će tabela, ako nije sačuvana na disku biti izgubljena. Nakon izlaska iz programa vraćamo se u 1—2—3 Access System. Izborom komande **Exit** vraćićemo se u DOS.

Ovim smo stekli osnovna znanja u radu sa Lotusom koja bi trebala da budu dovoljna za pravljenje jednostavnijih tabela. Prije prelaska na naprednije korištenje, razmotrimo detaljnije *Lotusove* funkcije i komande.

Primjer

Za vježbu napravićemo meni iz koga ćemo moći izabrati jednu od četiri kratke melodije. U opseg ćelija koji ćemo nazvati meni 1 upisaćemo slijedeće:

1. zvuk	2. zvuk	3. zvuk	4. zvuk
Beep 0123	Beep 1230	Beep 2301	Beep 3012
{BEEP 0}	{BEEP 1}	{BEEP 2}	{BEEP 3}
{BEEP 1}	{BEEP 2}	{BEEP 3}	{BEEP 0}
{BEEP 2}	{BEEP 3}	{BEEP 0}	{BEEP 1}
{BEEP 3}	{BEEP 0}	{BEEP 1}	{BEEP 2}

Makro koji će inicijalizirati meni, nazovimo ga **b,** mogao bi se sastojati od slijedećih komandi:

{MENUCALL meni1}	poziva meni na lokaciju meni1
{BRANCH\b}	skače na početak makroa

Pritiskom na Alt-B na kontrolnom panelu će se pokazati opcije i komentar za tekuću opciju. Opcija se može izabrati pomjeranjem pokazivača ili pritiskom na prvo slovo opcije.

Iz ove petlje može se izaći pritiskom na **Ctrl-Break i Esc.**

Help

Kada se pravi tabela koju treba da koristi još neko osim vas, zgodno je korisnicima obezbijediti pomoćne ekrane sa objašnjenjima **(help screens).**

Za to je potrebno rezervisati dio tabele za tekst objašnjenja. Tekst se unosi u vidu dugih labela.

Najjednostavniji makro treba da obezbijedi skok na lokaciju sa tekstom, da sačeka dok korisnik ne pritisne neku tipku, a zatim da se vrati na radni dio tabele. Složenije varijante mogu da uključe rad sa prozorima, a mogu se urediti i kao opcije menija sa više nivoa. Ovom makrou se obično daje ime H, tako da se pomoć dobija pritiskom na Alt-H.

Primjer

Pretpostavimo da smo lokacije od A100 pa na niže rezervisali za tekst pomoćnih objašnjenja. Makro za poziv tog teksta bi mogao da se sastoji od slijedećih komandi:

{GOTO} A100	skok na lokaciju teksta
{GET} A99	čeka se pritisak na tipku
{HOME}	vraća se na početak tabele
{GOTO} B8	skok na prvu ćeliju sa podacima u tabeli

Slide Show

Često je potrebno rezultate proračune u tabeli prikazati i grafički. To poboljšava preglednost a i djeluje dosta efektno. Naročito je korisno kod analiza tipa „šta ako", kada je moguće odmah vidjeti zavisnost pojedinih parametara. Da bismo iskoristili ova mogućnosti, potrebno je za naš model kreirati nekoliko karakterističnih grafova, neka ih bude tri. Grafovima ćemo komandom **/Graph Name Create,** dati neka imena.

U toku rada biće nam potrebno da, nakon svake promjene podataka, pregledamo sve grafove, jedan po jedan.

Makro za tu svrhu mogao bi izgledati otprilike ovako:

/GNU	komande /Graph Name Use
~	izbor prvog grafa
~{RIGHT}~	izbor drugog grafa
~{RIGHT} {RIGHT}~	izbor trećeg grafa
q	izbor opcije Quit

Naravno, nema nikakvog ograničenja da se radi i sa više od tri grafa. Samo je potrebno malo prilagoditi makro.

Baza podataka

Dio tabele može se tretirati i kao baza podataka. Bazu sačinjavaju kartoni u vidu redova. Svaki karton se sastoji od polja, koja su u ovom slučaju kolone. U njih se upisuju odgovarajući podaci. Kartone možemo zamisliti i kao kartice, a polja kao rubrike na njima. Baza bi u tom slučaju bila kartoteka.

Pri radu sa bazom imamo mogućnost da dodajemo nove, ili da brišemo stare kartone, da ih sortiramo po nekom ključu, da pronalazimo i/ili izdvajamo kartone sa određenim sadržajem itd.

Razmotrimo malo detaljnije osnovne operacije u radu sa bazom podataka.

{WRITELN string}

Upisuje stringu u otvorenu datoteku od pozicije pokazivača. Pri tome na kraj stringa dodaje znakove CR i LF

/X komande

Ova grupa komandi je zadržana zbog kompatibilnosti sa ranijim verzijama *Lotusa* (Release 1.xx) tako da stari korisnici ne moraju iznova pisati makroe. Slijedeća tabela pokazuje funkcionalnu ekvivalenciju sa novim komandama

/X komanda	Ekvivalent
/X*uslov*~	{IF}
/XG*lokacija*~	{BRANCH}
/XC*lokacija*	{ime_potprograma}
/XR	{RETURN}
/XQ	{QUIT}
/XL*poruka* lokacija	{GETLABEL}
/XN*poruka* lokacija	{GETNUMBER}
/XM*lokacija*	{MENUBRANCH}

Kutak za programere

Sada bi se moglo reći da je završen opis svih komandi. Možda su neki opisi isuviše kratki pa nekome neće biti dovoljni, međutim, već je rečeno da ovaj prilog ne može biti prava zamjena za originalno uputstvo.

Nakon što smo upoznali sve komande *Lotusa* razmotrimo nekoliko ideja za primjenu tog znanja u pravljenju sopstvenih aplikacija. Ma koliko *Lotus* bio jednostavan za korištenje, neke poslove je ipak zgodnije automatizovati, nego ih stalno raditi u interaktivnom načinu. Pravljenje aplikacija sem toga omogućava da i neobučeni vlasnici računara *Lotus* koriste za svoje specifične potrebe, ne razmišljajući pri tome o samom programu, niti kako je sistem modeliran. Za kreiranje aplikacija poslužićemo se, naravno, makroima. Gotovo je nemoguće navesti sve moguće primjene makroa, jer se one razlikuju od aplikacije do aplikacije, ali postoji nekoliko karakterističnih primjera na kojima ćemo se malo više zadržati.

Meniji

Uobičajeni način za komunikaciju korisnika sa programom je preko menija. Korisniku se ponudi nekoliko opcija, od kojih on izabere jednu, a zatim se izvrši odgovarajuća procedura. Osim toga ovaj način je i dosta siguran od grešaka.

Lotus je i korisnicima ostavio mogućnost da naprave menije po ugledu na originalni komandni linijski meni. Meniji mogu imati do 8 (osam) opcija sa objašnjenjima za svaku. Opcije se pojavljuju u drugoj, a objašnjenje u trećoj liniji kontrolnog panela.

Da bi se kreirao meni, neku oblast tabele treba urediti na slijedeći način

Opcija 1	Opcija 2
Komentar 1	Komentar 2

Opcija i komentara može biti, kao što je već rečeno, osam. Oni se upisuju u susjedne kolone. Između njih se ne smiju ostavljati prazni redovi. Ispod komentara, na mjestu tačkica, takođe bez preskakanja redova upisuju se makro komande. One predstavljaju program koji opslužuje odgovarajuću opciju iz menija.

Može se kreirati više menija za istu tabelu. Meniji se mogu gnijezditi, to jest opcija iz jednog menija može pozivati drugi meni itd. Meni se poziva komandama **{MENUBRANCH lokacija}** i/ili **{MENUCALL lokacija}**. Lokacija je koordinata ćelije u gornjem lijevom uglu meni oblasti. Umjesto koordinate bolje je upisati ime oblasti, naravno, pod uslovom da smo oblast prethodno imenovali komandom **/Range Name Create**.

Funkcije

Lotus raspolaže bogatim setom ugrađenih funkcija koje zadovoljavaju i najspecifičnije prohtjeve. Prema svojoj namjeni funkcije bi se mogle podijeliti u nekoliko osnovnih grupa

- — opšte matematičke funkcije
- — finansijske funkcije
- — statističke funkcije
- — trigonometrijske funkcije
- — logičke funkcije
- — tabelarne funkcije
- — funkcije za obradu grešaka
- — funkcije sa datum i vrijeme
- — string funkcije

Sve počinju znakom @ (ludo a) i mogu biti bez argumenata, ili sa njima. Argumenti se međusobno odvajaju zarezima. Kao argumenti mogu se pojaviti koordinate ćelija, blokovi ćelija, brojevi itd. Funkcije takođe može imati drugu funkciju za argument, ako ova daje odgovarajući izlaz. Na ovakvo gniježđenje funkcija nema nikakvih dodatnih ograničenja — jedino treba paziti da zagrade budu uravnotežene, tj. da imamo jednak broj otvorenih i zatvorenih zagrada, jer je to najčešća greška pri pisanju takvih formula.

U nastavku će funkcije biti obrađene po grupama, sa kraćim opisima i sintaksom.

Opšte matematičke funkcije

Ove funkcije namijenjene su za opšte matematičke operacije. Uglavnom ne zahtijevaju šira objašnjenja. Argumenti funkcija su numerički. To mogu biti brojevi, ali su češće to numeričke ćelije (ćelije čiji je sadržaj broj).

@abs(x)

Apsolutna vrijednost. Vraća apsolutnu vrijednost broja (numeričke ćelije)

@acos(x)

Arkus kosinus. Vraća vrijednost u radijenima (od 0 do Pi) ugla čiji je kosinus x. Ako x nije u opsegu od −1 do 1 vraća ERR

@asin(x)

Arkus sinus. Vraća vrijednost u radijanima (od −Pi/2 do Pi/2) ugla čiji je sinus x. Ako x nije u opsegu od −1 do 1 vraća ERR

@atan(x)

Arkus tangens. Vraća vrijednost u radijanima (od −Pi/2 do Pi/2) ugla čiji je tangens x.

@atan2(x,y)

Računa arkus tangens količnika y/x i vraća vrijednost u radijanima ugla čiji je tangens y/x. Pri tome vodi računa o predznacima x i y, tako da pokriva sva 4 kvadranta (ugao u opsegu od −Pi do Pi).

@cos(x)

Trigonometrijska funkcija kosinus. X je ugao u radijanima.

@exp(x)

Eksponencijalna funkcija, izračunava x-tu potenciju broja e

@int(x)

Cijeli broj. Vraća cijeli dio broja (ćelije).

@ln(x)

Prirodni logaritam — logaritam sa osnovom e

@log(x)

Dekadni logaritam — logaritam sa osnovom 10

@mod(x,y)

Ostatak dijeljenja x sa y

@pi

Broj Pi. Vraća vrijednost 3.141592653589794

- **rand**
  Slučajni broj. Vraća slučajan broj u opsegu između 0 i 1. Ne koristi argument.

- **round(x,n)**
  Zaokruživanje. Broj x zaokružuje na n decimalnih mjesta. I x i n mogu biti sadržaj ćelija.

- **sin(x)**
  Trigonometrijska funkcija sinus. X je ugao u radijanima.

- **sqrt(x)**
  Kvadratni korijen. Računa kvadratni korijen pozitivnog broja (numeričke ćelije).

- **tan(x)**
  Trigonometrijska funkcija tangens. X je ugao u radijanima.

Finansijske funkcije

Namjena

Korišćenjem ovih funkcija mogu se znatno olakšati i ubrzati proračuni u oblasti finansija (knjigovodstva). One zamjenjuju čitav niz elementarnih računskih operacija koje su potrebne da bi se napisale standardne formule za ovakve proračune. Argumenti su in numerički.

- **cterm(kamata, buduća_vrijednost, sadašnja_vrijednost)**
  Vraća broj potrebnih vremenskih perioda da neka sadašnja vrijednost uz navedenu kamatu dostigne zadanu buduću vrijednost.

- **ddb(nabavna_cijena, krajnja_cijena, trajanje, period)**
  Izračunava pad vrijednosti osnovnog sredstva za specificirani vremenski period koristeći „double-declining balance" metodu.
  Drugim riječima, izračunava koliko je u kojem periodu potrebno izdvojiti za amortizaciju osnovnog sredstva. Po ovoj metodi pad vrijednosti je brži u početnim godinama korišćenja.
  Nabavna_cijena je cijena po kojoj je osnovno sredstvo nabavljeno, krajnja_cijena je cijena osnovnog sredstva po isteku vijeka trajanja, a period predstavlja vrijeme za koje se izračunava pad vrijednosti.

- **fv(uplata, kamata, period)**
  Buduća vrijednost. Izračunava buduću vrijednost fiksnih uplata za određeni broj vremenskih perioda i fiksnu kamatu.

- **irr(x, opseg)**
  Interna stopa rentabiliteta (Internal Rate of Return). Vraća približnu internu stopu rentabiliteta za seriju ravnomjerno raspoređenih uplata, počevši od pretpostavljene vrijednosti x. Ako se za 20 interacija ne pojavi konvergencija unutar 0.000001, prekida se izvršenje i program vraća poruku ERR. U tom slučaju treba pokušati sa drugom pretpostavljenom vrijednošću. Opseg mora početi sa negativnom vrijednošću (investicija), a može sadržati pozitivne (uplate) i negativne (isplate) vrijednosti.

- **npv (kamata, opseg)**
  Neto sadašnja vrijednost. Izračunava sadašnju vrijednost budućih uplata u fiksnim vremenskim periodima i fiksnom kamatnom stopom. Uplate su navedene u opsegu.
  Ovim se, na primjer, može izračunati opravdanost neke investicije. Jako što će se sadašnji troškovi uporediti sa predviđenim budućim prihodima. Time, u stvari, poredimo da li bismo više zaradili da novac, uz navedenu kamatnu stopu, držimo na banci ili da ga uložimo u investiciju.

- **pmt (glavnica, kamata, period_vraćanja)**
  Otplata zajma. Izračunava ratu kredita na osnovu glavnice, kamate i broja rata (perioda vraćanja).

- **pv (uplata, kamata, period)**
  Sadašnja vrijednost. Vraća sadašnju vrijednost fiksnih budućih uplata uz fiksnu kamatu u navedenom periodu.

- **rate (buduća_vrijednost, sadašnja_vrijednost, period)**
  Izračunava kamatnu stopu potrebnu da sadašnja vrijednost dostigne buduću vrijednost u navedenom vremenskom periodu.

{LET lokacija, sadržaj}

Upisuje sadržaj u navedenu lokaciju. Sadržaj može biti broj ili string. Za određivanje tipa mogu se koristiti kvalifikatori value i string, u protivnom *Lotus* će probati da sam odredi tip.

{PUT opseg, kolona, red, sadržaj}

Smješta sadržaj u ćeliju na presjeku navedene kolone i reda u specificiranom opsegu. Kolone i red su brojevi i označavaju relativne pozicije počevši od nule.

{RECALC lokacija, <uslov>, <broj_iteracija>}

Rekalkuliše formule u specificiranom opsegu red po red. Može se iskoristiti za ubrzanje rada sa velikim tabelama kada nije potrebno rekalkulisati čitavu tabelu već samo neke dijelove. <uslov> i <broj_iteracija> su opcionalni parametri. Nakon svake rekalkulacije provjerava se da li je zadovoljen <uslov> pa ako nije, rekalkulacija se ponavlja. Broj ponavljanja može se ograničiti zadavanjem argumenta <broj_iteracija>. Svakom rekalkulacijom ovaj broj se smanjuje za jedan sve dok ne dostigne nulu. Ako su zadane oba argumenta, rekalkulacija se prekida kada bilo koji dostigne vrijednost FALSE (logička nula).

{RECALCCOL lokacija, <uslov>, <broj_iteracija>}

Isto što i {RECALC} samo što se rekalkulacija vrši po kolonama.

Rad sa datotekama

{CLOSE}

Zatvara datoteku otvorenu komandom {OPEN}.

{FILESIZE lokacija}

Određuje broj bajta u trenutno otvornoj datoteci i taj broj upisuje na navedenu lokaciju.

{GETPOS lokacija}

Određuje poziciju pokazivača u datoteci i taj broj upisuje na navedenu lokaciju. Početne pozicije pokazivača je 0 a ne 1.

{OPEN datoteka, način_pristupa}

Otvara navedenu datoteku za čitanje, pisanje ili oboje, u ovisnosti od navedenog argumenta. Moguće vrijednosti su R (Read), W (Write) i M (Modify). Ime datoteke može sadržati i oznaku jedinice i put ali ne smije biti duže od 64 karaktera. Može se navesti direktno, kao referenca na ćeliju koja sadrži string, ili kao operacija koja kao rezultat ima string.

{READ broj_znakova, lokacija}

Čita željeni broj znakova (maksimalno 240) iz trenutno otvorene datoteke i smješta ih na navedenu lokaciju kao labelu. Znakovi se čitaju od trenutne pozicije pokazivača u datoteci.

{READLN lokacija}

Čita red znakova iz trenutno otvorene datoteke i smješta je na navedenu lokaciju. Red počinje od pozicije pokazivača i završava se znakovima CR (Carriage Return) i LF (Line Feed).

{SETPOS pozicija}

Postavlja pokazivač u datoteci na željenu poziciju. Prvi znak je na poziciji 0, drugi na poziciji 1 itd.

{WRITE string}

Upisuje stringu u otvorenu datoteku od pozicije pokazivača. String se može navesti direktno, može biti sadržaj ćelije ili rezultat neke operacije.

Lokacija brojač označava ćeliju koja će se koristiti kao brojač, a početak lokaciju potprograma koji se izvršava u petlji.

{FORBREAK}

Prekida izvršenje {FOR} petlje i izvršenje prebacuje na prvi znak iza {FOR} komande. Koristi se samo u potprogramu koji se izvršava u petlji, u protivnom izaziva grešku.

{IF uslov}

Ako je uslov tačan izvršavaju se komande u istoj ćeliji u protivnom, u ćeliji ispod.

{ONERROR lokacija _ skoka, < lokacije _ poruke>}

Ukoliko se u toku izvršavanja makroa javi greška, program nastavlja izvršenje od lokacije _ skoka. Lokacija _ poruke je opcionalan parametar i u nju se smješta poruka koja bi se inače pojavila na dnu ekrana. Ako se ovaj parametar ne navede, nema načina da se sazna o kakvoj se grešci radi. Komanda je aktivna sve dok se ne naiđe na drugu {ONERROR} komandu ili dok se ne pojavi greška. Zato se obično na lokaciju _ skoka stavlja nova {ONERROR} komanda.

{QUIT}

Prekida izvršenje makroa i kontrolu vraća tastaturi.

{RESTART}

Nastavlja izvršenje potprograma ali briše stek, tako da se nailaskom na {RETURN} izvršenje ne vraća makrou iz koga je potprogram pozvan već se okončava.

{RETURN}

Povratak iz potprograma. Nije neophodno da se koristi na kraju potprograma.

Manipulacija podataka

{BLANK lokacija}

Briše ćelije na navedenoj lokaciji (opsegu).

{CONTENS odredište, izvorište, < širine >, < format >}

U odredišnu ćeliju upisuje kao string numeričku vrijednost izvorišne ćelije. <širina> i <format> su opcionalni numerički parametri koji čine da izvorišna ćelija izgleda kao da ima navedenu širinu i/ili format. Format je kodiran prema sljedećoj tabeli:

Kod	Format
0 do 15	Fixed sa 0 do 15 decimalnih mjesta
16 do 32	Scientific sa 0 do 15 mjesta
33 do 47	Currency sa 0 do 15 mjesta
48 do 63	% sa 0 do 15 mjesta
64 do 79	, (comma) sa 0 do 15 mjesta
112	+/−
113	General
114	D1 (DD—MMM—GG)
115	D2 (DD—MMM)
116	D3 (MMM—GG)
121	D4 (varira u odnosu na konfiguraciju)
122	D5 (varira u odnosu na konfiguraciju)
119	D6 (HH MM SS AM/PM)
120	D7 (HH MM AM/PM)
123	D8 (varira u odnosu na konfiguraciju)
124	D9 (varira u odnosu na konfiguraciju)
117	Text (prikazuje formulu kako je napisana)
118	Hidden (ne prikazuje sadržaj ćelije)
127	Predefinisani numerički format za tabelu

@ sln (nabavna_cijena, krajnja_cijena, period)

Izračunava pad vrijednosti osnovnog sredstva za specificirani vremenski period koristeći „straight-line" metodu. Kod ove metode vrijednost ravnomjerno opada za svaki vremenski period u toku vijeka trajanja.

@ syd (nabavna_cijena, krajnja_cijena, trajanje, period)

Izračunava pad vrijednosti osnovnog sredstva za specificirani vremenski period koristeći „sum-of-the-year's digits" metodu. Po ovoj metodi pad vrijednosti je brži u početnim godinama korištenja.

@ term (rata, kamata, buduća_vrijednost)

Izračunava period potreban da se, uz konstantnu kamatnu stopu i ratu koja se uplaćuje, dostigne neka buduća vrijednost.

Statističke funkcije

Kada je nad nekim opsegom numeričkih ćelija potrebno uraditi osnovne statističke obrade koristi se ova klasa funkcija.

@ count (opseg)

Brojač. Vraća broj svih nepraznih ćelija u opsegu.

@ sum (opseg)

Suma. Vraća sumu svih numeričkih ćelija u zadanom opsegu. @ sum (A1 . . A10) daje zbir numeričkih ćelija u koloni A u redovima 1 do 10.

@ avg (opseg)

Prosjek. Vraća srednju vrijednost numeričkih ćelija u zadanom opsegu.

@ min (opseg)

Minimum. Pronalazi minimalni broj u zadanom opsegu.

@ max (opseg)

Maksimum. Pronalazi maksimalni broj u zadanom opsegu.

@ std (opseg)

Standardna devijacija. Izračunava standardno odstupanje brojeva u zadanom opsegu. Prazne ćelije se zanemaruju.

@ var (opseg)

Varijansa. Izračunava varijansu brojeva u zadanom opsegu. Prazne ćelije se zanemaruju.

Logičke funkcije

Ove funkcije se koriste u slučajevima kada je potrebno odlučivanje, odnosno kada je potrebno svesti numeričke vrijednosti na logičke. Pri tome sve vrijednosti različite od nule svode se na istinu (logičku jedinicu), a samo je nula pogrešno (logička nula).

@ false

Pogrešno. Vraća logičku nulu.

@ true

Tačno. Vraća logičku jedinicu.

@ if (uslov, izraz1, izraz2)

Odlučivanje. Ako je uslov tačan, izvršava se x, a ako je pogrešan izvršava se y. U uslovu se mogu koristiti razni logički operatori. Ti operatori su sljedeći:

=	— jednako
<	— manje
<=	— manje ili jednako
>	— veće
>=	— veće ili jednako
<>	— nije jednako
#NOT#	— logičko NE
#AND#	— logičko I
#OR#	— logičko ILI

@ isna (x)

Vraća 1 ako je x=NA (Not Available). Koristi se za kontrolu grešaka

@ iserr (x)

Vraća 1 ako je x=ERR (Error). Koristi se za kontrolu grešaka.

@ isnumber (x)

Vraća 1 ako je ćelija x numerička, u protivnom vraća 0

@ isstring (x)

Vraća 1 ako ćelija x sadrži string u protivnom vraća 0.

Specijalne funkcije

U ovu grupu smještane su funkcije za razne specijalne primjene uglavnom za pristup ćelijama ne tabeli

@@ (x)

Vraća vrijednost ćelije na koju pokazuje ćelija x. Referenca se navodi kao labela (string)
Primjer @@(D9) vraća vrijednost 12.35 ako ćelija D9 sadrži labelu B2 a ćelija B2 ima vrijednost 12.35

@cell (atribut, opseg)

Vraća odgovarajući atribut za ćeliju. Atribut koji se zadaje je string i mora biti zatvoren u navodnicima. Takođe se može koristiti referenca na ćeliju koja sadrži atributski string. U zavisnosti od atributa mijenja se i vrijednost koju vraća formula. U sljedećoj tabeli dati su atributi i odgovarajući rezultati

Atribut	Rezultat
„adress"	vraća adresu ćelije
„row"	vraća broj reda (1—8192)
„col"	vraća broj kolone (1—256)
„contens"	vraća sadržaj ćelije
„type"	vraća tip podatka u ćeliji — b ako je ćelija prazna — v ako ćelija ima numeričku vrijednost — a ako ćelija sadrži string ili labelu
„prefix"	vraća prefiks labele — ' ako je labela lijevo poravnata — " ako je labela desno poravnata — ^ ako je labela centrirana — blanko ako ćelija ne sadrži labelu
„protect"	vraća status zaštićenosti ćelije — 1 ako je ćelija zaštićena — 0 ako ćelija nije zaštićena
„width"	širina kolone (1—240) format i tip podatka u ćeliji — F0—F15 brojevi sa fiksnim decimalama — S1—S15 brojevi u naučnom formatu — G brojevi u opštem formatu — P1—P15 brojevi u procentnom formatu — D1 datum DD—MMM—GG — D2 datum DD—MMM — D3 datum MMM—GG — D4 datum MM/DD/GG, DD/MM/GG, DD MM GG ili GG—MM—GG — D5 datum MM/DD DD/MM DD MM ili MM—DD — D6 vrijeme HH MM SS AM/PM — D7 vrijeme HH MM AM/PM — D8 vrijeme HH MM SS 24hr — D9 vrijeme HH MM 24 hr — T tekst format — H skriveni (Hidden) format — blanko za praznu ćeliju

Kod ove formule mora se zadati opseg čak i ako se navodi samo jedna ćelija. Ako opseg sadrži više od jedne ćelije podrazumijeva se ćelija u gornjem lijevom uglu opsega.

@ cellpointer (atribut)

Vraća vrijednost atributa za tekuću ćeliju (ćeliju na kojoj se nalazi pokazivač) u trenutku rekalkulacija

{WINDOWSOFF}

Zamrzava prikaz na ekranu izuzev kontrolnog panela. Ubrzava izvršenje makroa, pošto se ništa ne ispisuje na ekran

{WINDOWSON}

Vraća ekran u normalni način rada.

Rad sa tastaturom

{?}

Privremeno prekida izvršenje makroa i očekuje ulaz sa tastature. Da bi makro nastavio sa izvršenjem iza {?} treba da stoji ~ (tilda) kao oznaka za Return

{BREAKOFF}

Onemogućuje BREAK tipku u toku izvršenja makroa

{BREAKON}

Ponovo aktivira BREAK tipku i omogućava prekid izvršenja makroa.

{GET lokacija}

Očekuje unos jednog karaktera sa tastature i smješta ga na lokaciju. Ako je kao lokacija navedeno ime opsega, koristi se ćelija u gornjem lijevom uglu

{GETLABEL prompt, lokacija}

Ispisuje prompt i očekuje unos stringa koga smješta na navedenu lokaciju

{GETNUMBER prompt, lokacija}

Ispisuje prompt i očekuje unos broja koga smješta na navedenu lokaciju.

{LOOK lokacija}

Ispituje da li je u toku izvršenja makroa pritisnuta neka tipka. Ako jeste smješta odgovarajući znak na lokaciju, u protivnom briše lokaciju

{MENUBRANCH lokacija}

Privremeno zaustavlja izvršenje i omogućava izbor opcije iz menija definisanog na navedenoj lokaciji. U skladu sa izborom izvršava odgovarajući makro

{MENUCALL lokacija}

Privremeno zaustavlja izvršenja i omogućava izbor opcije iz menija definisanog na navedenoj lokaciji. U skladu sa izborom izvršava odgovarajući makro kao potprogram

{WAIT vrijeme}

Zaustavlja izvršenje makroa dok se ne dostigne vrijeme navedeno kao redni broj

Kontrola izvršenja programa

{BRANCH lokacija}

Nastavlja izvršenje makroa od navedene lokacije. Ekvivalent komandi GOTO u drugim jezicima.

{DEFINE lokacija1: tip1, lokacija 2: tip 2, ... }

Određuje lokacije i tipove argumenata koji će biti prosljeđeni potprogramu. Tip može biti value za brojeve ili string za stringove. Ako se izostavi podrazumijeva se string

{DISPATCH lokacija}

Nastavlja izvršenja makroa od ćelije čije je adresa sadržana na lokaciji. Ovo je neka vrsta indirektnog adresiranja

{FOR brojač, start, stop, korak, početak}

Ekvivalent FOR-NEXT petlje. Brojač i početak su lokacije a ostali parametri mogu biti brojevi ili lokacija

Planiranje makroa je stvar konkretne aplikacije. U početku je to uglavnom stvar iskustva kada korisnik primijeti da često ponavlja neke stvari pa odluči da sebi olakša život. Provjeravanje makroa manuelnim izvršavanjem, takođe nije problem. Međutim problemi nastupaju kada treba tu sekvencu zapisati. Zadržimo se zato malo više na drugom koraku.

U makroima na primjer često ima potrebe za naznačavanjem kretanja po tabeli. Međutim kako naznačiti korištenje kurzorskih strjelica, ili funkcijskih tipki kada one nemaju odgovarajućeg karaktera koji bi se mogao ukucati kao dio makroa. Za te i slične potrebe koriste se specijalne oznake. U tabeli su sa lijeve strane prikazane makro sekvence, a sa desne opis djelovanja odnosno odgovarajuća tipka.

Makro komanda	Dejstvo
~	Return
{down}	Kurzorska strjelica dole
{up}	Kurzorska strjelica gore
{left}	Kurzorska strjelica lijevo
{right}	Kurzorska strjelica desno
{home}	Home
{end}	End
{pgup}	PgUp
{pgdn}	PgDn
{bigleft}	Jedan ekran lijevo
{bigright}	Jedan ekran desno
{edit}	EDIT (F2)
{name}	NAME (F3)
{abs}	ABS (F4)
{goto}	GOTO (F5)
{window}	WINDOW (F6)
{query}	QUERY (F7)
{table}	TABLE (F8)
{calc}	CALC (F9)
{graph}	GRAPH (F10)
{escape} ili {esc}	Esc
{backspace} ili {bs}	BackSpace (brisanje ulijevo)
{delete} ili {del}	Del (koristi se samo u EDIT modu)
{~}	Tilda
{{} i {}}	Zagrade { i }

Ako se želi naznačiti više uzastopnih korištenja iste tipke u zagradu se može staviti i broj ponavljanja. Na primjer {up 4} izaziva pomjeranje pokazivača za četiri ćelije na gore.

Sve u svemu na ovom nivou makro mogućnosti *Lotusa* ne odskaču mnogo od mogućnosti specijalnih makroprocesora — programa kao što su *SuperKey* ili *ProKey*. Čak su i lošije jer ne postoji mogućnost direktnog „animanja" makroa u toku obavljanja operacije koja se automatizuje. Međutim to je tek dio mogućnosti.

Makro komandni jezik

Za naprednije korisnike na raspolaganju je pravi makro programski jezik sa komandama kao što su {IF}, {BRANCH}, {RETURN}, ... sa pozivima potprograma kojima se mogu prosljeđivati parametri itd. To je veliko poboljšanje u odnosu na prethodne verzije programa (Release 1.xx) koje su imale samo tzv. /X makro komande.

Komande mogu stajati same ili imaju jedan ili više argumenata. Argumenti su od komande odvojeni blankom a međusobno zarezima, ili tačka zarezom. Komandom **/Worksheet Global Default Other International Punctuation** mogu se definisati i drugi separatori. Komanda i argumenti zatvoreni su u vitičaste zagrade. Moguća su četiri vrste argumenata: brojevi, stringovi, lokacije i logički izrazi. U daljem tekstu biće dati kraći opisi komandi podijeljenih prema funkcionalnosti u odgovarajuće grupe.

Kontrola ekrana

{BEEP broj}

Aktivira zvučnik. Numerički argument je opcionalan. Može biti 1, 2, 3, ili 4. Ako se ne navede podrazumijeva se 1.

{INDICATE string}

Mijenja indikator moda u gornjem desnom uglu ekrana. String argument je opcionalan. Ako se stavi prazan string indikator se uklanja.

{PANELOFF}

Onemogućava promjene kontrolnog panela u toku izvršavanja makroa.

{PANELON}

Omogućava promjene kontrolnog panela.

@ choose (x, v1, v2, . . . ,vn)

Izbor. Na osnovu vrijednosti x bira vrijednost iz navedene liste. Na primjer za x=1 vratiće vrijednost v1 za x=2 vrijednost v2 itd.

@ cols (opseg)

Vraća broj kolona u navedenom opsegu. Korisno je u slučajevima kada se umjesto koordinata koristi imenovani opseg.

@ rows (opseg)

Vraća broj redova u navedenom opsegu. Korisno je u slučajevima kada se umjesto koordinata koristi imenovani opseg.

@ hlookup (x, opseg, ofset)

Horizontalno traženje po tabeli. Obično se koristi u slučajevima kada radimo sa koeficijentima koji imaju samo tabelarnu zavisnost od ulazne vrijednosti (na primjer poreske stope za neme osnovice). Ovdje se pod tabelom podrazumijeva uređen opseg ćelija. U prvom redu nalaze se vrijednosti koje se ispituju (x). One moraju biti jedinstvene i poredane u rastućem redoslijedu. Ako se tražena vrijednost nalazi negdje između dvije tabelarne, usvaja se manja i ta kolona postaje tekuća. Ako je vrijednost x manja od prve tabelarne vrijednosti funkcija vraća ERR.

U ostalim redovima nalaze se razni koeficijenti vezani za te vrijednosti. Ofset naznačava iz kog će se reda ispod reda sa ulaznim podacima, čitati vrijednost koeficijenta tekuće kolone. Te vrijednost je izlaz funkcije. Ofset ne smije biti negativan niti smije preći granice naznačenog opsega.

Primjer: Ako imamo tabelu sa koeficijentima

	A	B	C	D
1	100	150	200	250
2	1.55	2.30	3.00	3.50
3	30	70	85	97
4				
5	178	1		

formula @ hlookup (A5, A1 ... D3, B5) vratiće vrijednost 2.30 a formula @ hlookup (200, A1 ... D3,2) vratiće vrijednost 85.

@ vlookup (x, opseg, ofset)

Vertikalno traženje po tabeli. Analogna je funkciji @ hlookup samo što se vrijednost x traži u prvoj koloni opsega, a ofset pokazuje iz koje se kolone čita vrijednost koeficijenta.

@ na

Vraća poruku da ćelija nije raspoloživa. Obično se koristi zajedno sa funkcijom @ if.

Primjer: @ if (A5<0, @ na, @ sqrt (A5)). U slučaju da je sadržaj ćelije A5 manji od nula prikazuje se poruka NA, u protivnom računa se kvadratni korijen.

@ err

Vraća poruku o grešci. Obično se koristi zajedno sa funkcijom @ if.

Primjer: @ if (A5<0, @ err, @ sqrt (A5)). U slučaju da je sadržaj ćelije A5 manji od nula prikazuje se poruka ERR, u protivnom računa se kvadratni korijen.

index (opseg, kolona, red)

Vraća vrijednost ćelije koja se nalazi na presjeku zadane kolone i reda u navedenom opsegu. Vrijednosti za kolonu i red su relativne i odnose se na lijevu i gornju granicu opsega. Tako prva kolona i prvi red opsega imaju vrijednost 0, slijedeći red i kolona vrijednost 1 itd.

Datumske funkcije

Namjena ovih funkcija su razne manipulacije sa datumima i vremenom kao i rad sa ugrađenim kalendarom i sistemskim časovnikom. Ugrađeni kalendar radi sa svim datumima između 1. januara 1900. do 31. decembra 2099. godine. Pri tome godina je predstavljena brojevima od 0 (1900.) do 199 (2099.), mjesec brojevima od 1 do 12 i dan brojevima od 1 do 31. Bez obzira na dozvoljene vrijednosti provjerava se logičnost datuma na osnovu ugrađenog kalendara. To znači da ne možemo zadati datum 30. 02. 1988. Da bi se mogle sprovoditi određene kalkulacije, svakom danu je dodijeljen redni broj. Slično je i sa vremenom. Dan predstavlja vrijednost 1. Ponoć ima vrijednost 0, podne 1/2 a neposredno pred ponoć dobija se vrijednost 0.99999.

Funkcije se mogu podijeliti u dvije grupe. Prva grupa generiše redne brojeve za datume i vrijeme, a druga te brojeve koristi kao ulazne parametre.

Funkcije koje generišu redne brojeve za datum i vrijeme

- **date (godine, mjesec, dan)**

Datum. Vraća redni broj dana koji odgovara navedenom datumu, počevši od 1. januara 1900. Tako 1. januar 1900. ima redni broj 1, 2. januar 1900. redni broj 2 itd.

- **datevalue (datumski string)**

Vraća redni broj dana. Za razliku od funkcije date koja kao ulaz koristi tri numeričke vrijednosti za mjesec, dan i godinu, ova funkcija koristi jedinstveni string koji može biti napisan u jednom od pet mogućih formata.

D1 DD—MMM—GG
D2 DD—MMM podrazumijeva se godina sistemskog kalendara
D3 MMM—GG podrazumijeva se prvi u mjesecu
D4 MM/DD/GG DD/MM/GG DD.MM.GG ili GG—MM—DD
D5 MM/DD DD/MM DD.MM ili MM—DD

- **day (datum)**

Na osnovu datuma (rednog broja) vraća redni broj dana u mjesecu.

- **now**

Vraća redni broj za tekući datum i vrijeme na osnovu sistemskog kalendara i časovnika.

- **time (sat, min, sek)**

Vraća redni broj za zadano vrijeme. Sat je broj u opsegu 0—23, a min i sek broj u opsegu 0—59.

- **timevalue (vremenski string)**

Vraća redni broj za zadano vrijeme. Za razliku od funkcije time koja kao ulaz koristi tri numeričke vrijednosti za sate, minute i sekunde, ova funkcija koristi jedinstveni string koji može biti napisan u jednom od četiri moguća formata.

D6 HH.MM.SS AM/PM
D7 HH.MM AM/PM
D8 HH.MM.SS HH.MM.SS ili HHhMMmSSs (24 sata)
D9 HH.MM HH.MM ili HHhMMm (24 sata)

Funkcije koje koriste redne brojeve za datum i vrijeme

- **day (datum)**

Vraća redni broj dana u mjesecu (1 do 31) na osnovu rednog broja datuma.

- **month (datum)**

Vraća redni broj mjeseca u godini (1 do 12) na osnovu rednog broja datuma.

- **year (datum)**

Vraća redni broj godine (0 do 199) na osnovu rednog broja datuma.

- **hour (vrijeme)**

Na osnovu rednog broja vremena vraća vrijednost sati između 0 (ponoć) i 23 ili 11 PM u zavisnosti od izabranog formata.

- **minute (vrijeme)**

Na osnovu rednog broja vremena vraća vrijednost minuta između 0 i 59.

- **second (vrijeme)**

Na osnovu rednog broja vremena vraća vrijednost sekundi između 0 i 59.

Funkcije za rad sa bazom podataka

Ove funkcije su ekvivalent statističkim funkcijama, samo što se primjenjuju na bazu podataka. Sve rade na isti način: koriste ulazni opseg (uo) i opseg kriterija (ok) da izdvoje odgovarajuće kartone iz baze, a ofset da odrede nad kojom će se kolonom (pojem) vršiti odgovarajuća statistička obrada.

\> — nastavak bloka
* — neposredno ispod se nalaze blanko karakteri trenutno nedefinisani, ali mogu biti dio podataka u ćelijama ispod

Može se formirati više format linija. To se na primjer radi kada su sa podacima uvezena i zaglavlja kolona. Tada se podaci u jednom redu (zaglavlje) razlikuju od podataka u ostalim redovima. Nakon što *Lotus* formira format liniju, izborom opcije **Edit**, mogu se napraviti korekcije. Nepotrebna format linija može se obrisati kao i svaki drugi red u tabeli.

Opcijom **Output-Range** treba naznačiti gornji lijevi ugao opsega u koji će se prebaciti podaci u formatiranom obliku, podijeljeni u zasebne kolone.

Izborom opcije **Go** izvršavamo komandu, **Reset** poništava sve postavljene redne parametre, a **Quit** nas vraća u READY mod.

/System

Omogućava „izlet" u DOS bez prekida rada sa *Lotusom*.

U prvoj verziji programa, za korisnike koji su radili na sistemima bez tvrdog diska, poruka da je disk sa podacima pun bila je vrlo neprijatna ako pri ruci nisu imali drugu formatiranu disketu. Sada takav problem više ne postoji jer se može za trenutak izaći u DOS i formatirati disketu, iskopirati datoteke, formirati direktorij ili uraditi nešto slično. Pri tome treba paziti da se ne pozove neki drugi program iz grupe tzv. eksternih DOS komandi kao što je na primjer PRINT. U *Lotus* se vraćamo uz pomoć komande EXIT.

/Quit

Komanda za prekid rada sa *Lotusom*. Svi podaci, ako nisu prethodno snimljeni na disk, biće izgubljeni.

Vježba čini majstora

Nakon ovog ne baš pretjerano interesantnog, ali svakako potrebnog nabrajanja funkcija i komandi stvari su vjerovatno mnogo jasnije. Već se mogao steći utisak šta sve *Lotus* može, a može mnogo.

Ovo znanje bi trebalo početnicima da bude dovoljno za izradu i korištenje sopstvenih tabela. Možda u početku neće baš sve ići glatko, ali ne treba zaboraviti da vježba čini majstora. Nadam se da će se uskoro pojaviti i prilozi u kojima će biti opisana rješenja nekih konkretnih problema.

Međutim, ovim priča o *Lotusu* nije gotova. Naprotiv! Zato, ako se ne zadovoljavate sa onim što ste do sada pročitali i želite da napravite korak dalje, razmotrimo neke načine naprednog korištenja ovog programa. Oni podrazumijevaju automatiziranje nekih operacija na tabeli, izradu i prikaz „slide show-a", izradu sopstvenih aplikacija sa menijima i helpom, jednom riječju programiranje tabele. Priča o tome je priča o makroima.

Moćno oruđe — makroi

Makro je neka vrsta programa sastavljenog od ekvivalenata pritisaka na tipke i komandi. Sadržaj makroa upisuje se u ćelije na tabeli. Svaki makro ima ime po kojem se poziva. Pozivom makroa, *Lotus* izvršava instrukcije od kojih je sastavljen. Nakon završetka izvršenja možemo normalno nastaviti rad.

Makroi omogućavaju da se automatizuje gotovo svaki posao koji možemo uraditi ručno. Time štede vrijeme i pojednostavljuju rad sa tabelom. Uobičajena primjena makroa je za:

— automatiziranje često korištenih komandi
— kucanje iste labele na više mjesta u tabeli
— izvršavanje procedure koja se često ponavlja, a koja zahtijeva seriju uzastopnih komandi
— pravljenje kompletnih aplikacija za nekoga ko nije „stručnjak" za *Lotus*

Razmotrimo i makroe u nekoliko koraka.

Osnove

Od čega se makro sastoji to smo već rekli. Taj sadržaj se upisuje u jednu, ili niz uzastopnih ćelija u jednoj koloni, u vidu labele bez obzira da li su to brojevi, formule, komande iz menija, ili ekvivalenti pritisaka na tipke. Kod izvršavanja, čita se sadržaj prve ćelije s lijeva na desno, a zatim se to ponavlja i sa ostalim ćelijama ispod, sve dok se ne naiđe na praznu ćeliju ili komandu koja označava kraj. Izvršenje se prekida i u slučaju greške u makrou.

Makro se kreira u tri koraka.

1. Planiranje makroa, manuelno izvršenje korak po korak i pažljivo zapisivanje svakog koraka.
2. Unos makroa u vidu labela u jednu ili više ćelija. Oblast za unos makroa obično se bira daleko od oblasti sa podacima, kako ne bi došlo do miješanja. Sadržaj svake ćelije može biti dug do 240 znakova, ali je zbog ispravki i preglednosti zgodno duge labele podijeliti na nekoliko kraćih.
3. Imenovanje makroa. Za to se koristi komanda **/Range Name Create**. *Lotus* prihvata samo slova A-Z sa obrnutom kosom crtom / (backslash) ispred kao ime makroa. Kao opseg se označava samo prva ćelija makroa. Preporučljivo je lijevo od te ćelije odmah upisati naziv makroa. Makro se poziva istovremenim pritiskom na MACRO tipku (Alt) i slovo koje predstavlja naziv. Ako želimo da se makro automatski izvrši svaki put kada učitamo tabelu, za ime trebamo odabrati \0 (nula). Ovaj makro nije moguće pozvati kasnije u toku rada sa tabelom. Zato, ako nam je potreban, trebamo mu dati i neko drugo ime.

U izlazni opseg **(Output Range)** smještaju se kartoni izdvojeni iz baze po nekom uslovu. Opseg se sastoji od reda sa nazivima polja koja želimo da prikažemo i odgovarajućeg broja redova ispod. Nazivi polja moraju biti isti kao i u bazi, ali mogu biti u proizvoljnom redoslijedu. Za opseg se može uzeti i samo red sa nazivima polja. Izdvojeni kartoni će zauzimati redove ispod bez ograničenja na dužinu. Pri tome treba paziti da ne dođe do preklapanja sa drugim podacima na tabeli.

Find opciju biramo kada želimo u bazi pronaći kartone koji zadovoljavaju uslov zadan u opsegu kriterija. Svi odgovarajući kartoni biće prikazani inverzno. Po kartonima se krećemo kurzorskim strelicama za gore i dole. Ako se pretražuju polja sa labelama mogu se koristiti i specijalni džoker karakteri. Znak? (upitnik) zamjenjuje jedan karakter, * (zvijezdica) grupu karaktera, a ako ispred niza znakova stoji ~ (tilda) kartoni u kojima se taj niz pojavljuje se isključuju.

Extract opcijom se služimo kada u izlaznom opsegu želimo da dobijemo kopije kartona koji zadovoljavaju zadani uslov.

Ako želimo da u izlazni opseg izdvojimo samo jedinstvene kartone, poslužićemo se opcijom **Unique.**

Delete opcijom iz baze se brišu svi kartoni koji zadovoljavaju uslov.

Reset briše sve postavljene parametre, a **Quit** nas vraća u READY mod.

Posljednju **/Data Query** komandu možemo ponoviti iz READY moda pritiskom na funkcijsku tipku **F9** (QUERY).

/Data Distribution

Omogućava analizu frekvencije raspodjele vrijednosti u opsegu. Tako možemo saznati koliko vrijednosti iz opsega pada u specificirane granice.

Prije izvršenja komande potrebno je da izaberemo dvije susjedne prazne kolone. U lijevu se upišu brojevi u rastućem redoslijedu. Oni predstavljaju vrijednosti granica za koje ćemo analizirati raspodjelu. U desnoj koloni nakon izvršenja komande pojaviće se vrijednosti frekvencije.

Po pozivu, komanda zahtjeva da se naznači opseg sa vrijednostima **(Value Range)**, a zatim i opseg sa granicama raspodjele **(Bin Range)**. Brojevi koji se pojave u desnoj koloni pokazuju koliko u opsegu sa vrijednostima ima vrijednosti koje su manje ili jednake odgovarajućoj graničnoj vrijednosti, ali veće od prethodne granične vrijednosti. Kolona sa frekvencijama ima i red više, gdje je prikazano koliko ima vrijednosti većih od posljednje vrijednosti **Bin** opsega.

/Data Matrix

Omogućava množenje i inverziju matrica.

Dijelove tabele možemo smatrati matricama i nad njima obavljati ove dvije osnovne matrične operacije. Pri tome važe pravila da možemo invertovati samo kvadratnu matricu, to jest matricu sa istim brojem redova i kolona, a kod množenja matrica da je broj kolona prve matrice jednak broju redova druge. Rezultati ovih operacija su opet matrice. U prvom slučaju rezultujuća matrica je istih dimenzija kao i ulazna, a u drugom slučaju dobija se matrica koja ima redova kao prva i kolona kao druga. Maksimalne dimenzije matrica su 90 redova sa 90 kolona. Matrica se zadaje kao opseg.

Data Regression

Omogućava sprovođenje regresione analize.

Računaju se vrijednosti koeficijenata i konstante za formulu koja povezuje jedan ili više setova nezavisnih varijabli sa setom zavisnih varijabli. Takođe, računa se i statistička sigurnost tih vrijednosti.

Ova komanda se koristi kada imamo nekoliko setova naoko nezavisnih varijabli a želimo da vidimo da li među njima ipak postoji neka zavisnost. Može se zadati maksimalno 16 (šesnaest) setova nezavisnih varijabli. One se zadaju kao X opseg. U Y opsegu nalazi se set zavisnih varijabli. Potrebno je navesti i izlazni opseg **(Output Range)** u kom će se pojaviti rezultati analize sa svim potrebnim komentarima.

Može se nacrtati i graf koji pokazuje aproksimacionu liniju. Za to je potrebno prvo proračunati očekivanu vrijednost Y po formuli

Y = koeficijent _ od _ X1*X1 + Koeficijent _ od _ X2*X2 + ... + konstanta

Vrijednosti X1, X2, ... su vrijednosti odgovarajućih setova nezavisnih varijabli, a koeficijenti i konstanta se mogu pročitati u izlaznom opsegu.

Data Parse

Konvertuje kolonu sa dugim labelama u nekoliko kolona sa labelama i brojevima. Vrlo korisna komanda kada se preuzimaju podaci u ASCII obliku komandom **/File Import Text.**

Za početak opcijom **Input-Column**, treba odrediti kolonu u kojoj se nalaze „uvezeni" podaci u vidu dugih labela.

Opcija **Format-Line** omogućava kreiranje ili mijenjanje formul linije. Ta linija pokazuje granice i vrstu podataka u budućim kolonama, tj. pokazuje kako će kolone sa dugima labelama biti podijeljene. Linija počinje znakom | (uspravna isprekidana crta), a umeće se ispod reda sa pokazivačem. U njoj se mogu pojaviti sljedeći simboli koji označavaju:

- L — početak bloka labela
- V — početak numeričkog bloka
- D — početak bloka sa datumima
- T — početak bloka sa vremenima
- S — karakteri ispod se preskaču i ne pojavljuju se u izlaznom opsegu (ne ubacuje se automatski, već ručno)

@dcount (uo, ofset, ok)

Broj kartone u datoteci koji ulaze u obradu.

@dsum (uo, ofset, ok)

Suma vrijednosti naznačenog numeričkog polja.

@davg (uo, ofset, ok)

Srednja vrijednost naznačenog numeričkog polja.

@dmin (uo, ofset, ok)

Minimalna vrijednost naznačenog numeričkog polja.

@dmax (uo, ofset, ok)

Maksimalna vrijednost naznačenog numeričkog polja.

@dstd (uo, ofset, ok)

Standardna devijacija zadanog numeričkog polja.

@dvar (uo, ofset, ok)

Varijansa zadanog numeričkog polja.

String funkcije

Ove funkcije će koristiti uglavnom napredniji korisnici u izradi sopstvenih aplikacija. Operacije sa stringovima su veoma dobro podržane. Na njima bi mu mogli pozavidjeti i pravi programski jezici. Stringovi mogu biti dugi do 240 znakova. Prvi znak u stringu ima poziciju 0.

@char (x)

Vraća znak čiji je ASCII kod x.

@code (string)

Vraća ASCII kod prvog znaka u stringu.

@exact (string1, string 2)

Vraća 1 ako su string 1 i string 2 identični, u protivnom vraća 0.

@find (uzorak, string, početne pozicije)

Vraća poziciju na kojoj se uzorak prvi put pojavljuje u stringu počevši od početne pozicije. Ako traženje nije uspješno vraća ERR.

@left (string, n)

Vraća prvih n znakova stringa.

@right (string, n)

Vraća zadnjih n znakova stringa.

@lenght (string)

Vraća dužinu stringa.

@lower (string)

Konvertuje sva velika slova u stringu u mala.

@upper (string)

Konvertuje sva mala slova u stringu u velika.

@mid (string, početna pozicije, n)

Isijeca n znakova iz stringa počevši od početne pozicije.

- **n (opseg)**
Vraća numeričku vrijednost ćelije u gornjem lijevom uglu opsega.

- **s (opseg)**
Vraća string vrijednost ćelije u gornjem lijevom uglu opsega

- **proper (string)**
Konvertuje sve riječi u stringu tako da im je prvo slovo veliko, a ostala mala
- **repeat (string, n)**
Vraća string ponovljen n puta
- **replace (string1, početna pozicija, n, string 2)**
Uklanja n znakova sa početne pozicije u stringu 1 i na to mjesto umeće string 2. Vrlo fleksibilna funkcija. Varijacijom parametara mogu se postići različiti efekti
- **string (x, n)**
Vraća broj x kao string sa n decimalnih mjesta.
- **value (string)**
Vraća numeričku vrijednost stringa koji izgleda kao broj
- **trim (string)**
Uklanja blanko znakova sa početka i kraja stringa

Komandni meni

Nakon funkcija neophodno je da upoznamo i komande.
Komandama se služimo kada je potrebno da obavimo operacije kao što su
— kopiranje, pomjeranje i brisanje podataka u tabeli
— čitanje sa ili smještanje podataka na disk
— štampanje izvještaja
— crtanje grafova
— rad sa bazom podataka

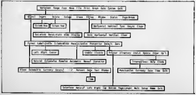
Sve komande smještene su u menije, koji se granaju u više nivoa. Glavni meni se poziva pritiskom na tipku sa znakom / (kosa crta). Za kretanje po meniju i izbor komande služimo se kurzorskim strelicama za lijevo i desno i tipkom **Enter**. Isti efekat se postiže pritiskom na prvo slovo komande. Tako smo, u našem primjeru, koristili komandu **Copy**.

Raspoložive komande prikazane su u drugoj liniji ekrana. U trećoj liniji pojavljuje se objašnjenje tekuće (osvijetljene) komande ili komande podmenija, ako takav postoji.

Rad sa komandama je lagan, jer su komande interaktivne. Zadavanje komande i njenih parametara može imati jedan, ili više koraka. U svakom koraku odgovarajući komentar objašnjava šta je potrebno uraditi — unijeti broj, takal oseg, ime datoteke i slično. U slučaju da pogriješimo ili se predomislimo, pritiskom na tipku **Esc** vraćamo se za jedan korak unazad.

U daljem tekstu biće obrađena komande po grupama. Uz objašnjenje biće date i shema grananja opcija unutar svake grupe.

/Worksheet



smještena u poljima tj. kolonama. Polja imaju svoje nazive i sadrže tačno određenu vrstu podataka. Ne smiju se, na primjer, ubacivati brojevi u polje koje sadrže labele. Svi zapisi u jednoj bazi imaju ista polja. Nazivi polja moraju se nalaziti u posebnim kolonama u prvom redu opsega koji je označen kao baza. Maksimalan broj polja je 256. Ne smije biti nikakvih suvišnih redova, praznih ili sa nekim linijama za odvajanje između naziva polja i prvog rekorda, kao ni između samih rekorda u bazi.

/Data Fill

Omogućava da se određeni opseg tabele ispuni brojevima u rastućem ili opadajućem opsegu. Za komandu je potrebno zadati opseg, startnu vrijednost, korak i krajnju vrijednost. Izborom predznaka koraka bira se rastući ili opadajući niz. Opseg se ispunjava počevši od ćelije u gornjem lijevom uglu na dole. Kada se ispuni jedna kolona prelazi se na drugu, sve dok se opseg ne ispuni ili dok se ne dostigne krajnja vrijednost. Ova komanda se može zgodno iskoristiti za vraćanje originalnog redoslijeda sortiranih slogova. Prije sortiranja treba slogove komandom **/Data Fill** numerisati. Brojevi se upisuju u kolonu uz opseg za sortiranje. Kod sortiranja treba voditi računa da sada i ta kolona uđe u opseg. Kada se želi vratiti stari redoslijed, izvrši se ponovno sortiranje, samo što se za primarni ključ uzme kolona sa brojevima.

/Data Table

Ova opcija skriva dvije vrlo moćne komande za analizu podataka. One omogućavaju da se neka formula automatski primijeni na različite ulazne podatke i rezultati prikažu u vidu tabele. To u mnogome olakšava razne analize, testove tipa šta-ako?, pripremu grafika, analizu podataka iz datoteke i slično.

/Data Table 1 kreira tabelu u kojoj su prikazane različite vrijednosti koje formula generiše svaki put kada se u njoj primijeni jedna vrijednost. Tabelom se može testirati i više formula, a tim što je u svim ista promjenjiva. Za tabelu je potrebno odrediti pravougaoni opseg ćelije i takozvanu ulaznu ćeliju. Ta ćelija je nezavisna od opsega i u formulama se koristi kao promjenjiva. Ćelija u gornjem lijevom uglu opsega ostavlja se slobodna. U istom redu, u kolona desno od te ćelije, upisuju se formule, a u istoj koloni, u redove ispod, upisuju se vrijednosti sa kojima će se formule testirati. U ostalim ćelijama biće prikazani rezultati. Komanda radi na taj način što redom ulaznoj ćeliji dodjeljuje vrijednosti prve kolone, a zatim proračunava formule. Ulazne vrijednosti mogu biti i labele, već u ovisnosti od funkcija koje se koriste.

Iskusniji korisnici će uz pomoć funkcija koje rade sa bazom podataka, kao što su e **dsum**, e **davg** (vidi funkcije) i slične, lako napraviti analiza podataka iz baze. U kolonu sa ulaznim vrijednostima treba upisati vrijednosti po kojima će se vršiti izdvajanje rekorda iz baze u opsegu kriterija, a za ulaznu ćeliju izabrati ćeliju iz opsega kriterija, neposredno ispod naziva polja po kom se vrši izdvajanje (rad sa bazom podataka biće detaljnije obrađen u posebnom poglavlju).

/Data Table 2 kreira tabelu sa rezultatima koji su posljedica promjene dvije varijable u formuli. U skladu sa tim, osim opsega, potrebno je odrediti i dvije ulazne ćelije. Sadržaj opsega takođe je nešto drugačiji. Ćelije u gornjem lijevom uglu, koje je u prethodnom slučaju bila prazna, sada je iskorištena za formulu a na mjesto formula upisan je set vrijednosti druge varijable. Set vrijednosti prve varijable upisuje se na uobičajeno mjesto — u prvu kolonu opsega. Rezultati za svaku kombinaciju ulaznih vrijednosti biće prikazani u ostalim ćelijama opsega. Ako su se ulazne ćelije izabrane ćelije koje već imaju neke vrijednosti, primjena komande neće uticati na njihovu promjenu.

Primjenom ove komande podaci u bazi mogu se analizirati po dva kriterija, na sličan način kao što je već opisano kod **/Data Table 1**.

Opcija **Reset** poništava već definisane parametre.

Još treba napomenuti da promjena podatake ili formula ne izaziva automatsko izvršenje komandi i rekalkulaciju, već je za to potrebno pritisnuti tipku **F8** (TABLE).

Još jedna caka: ćelije sa formulama mogu se komandom **/Range Format Text** prevesti u tekst format, što ne utiče na njihovu funkcionalnost, ali poboljšava preglednost tabele.

/Data Sort

Omogućava rearanžiranje kartona u bazi prema zadanim uslovima.

Opcija **Data-Range** koristi se za naznačavanje opsega koji će se sortirati. U opseg ne treba da uđu nazivi polja.

Opcijom **Primary-Key** određujemo naziv polja koje će biti korišteno kao primarni ključ za sortiranje. Kartoni mogu biti složeni u rastućem (Ascending), ili opadajućem (Descending) nizu.

Izborom opcije **Secondary-Key** može se zadati i sekundarni ključ, koji ima efekta ako se nađe više rekorda sa istim sadržajem primarnog ključa.

Opcija **Reset** služi za poništavanje prethodno zadatih parametara. **Go** se koristi za pokretanje procesa sortiranja, a **Quit** nas vraća u READY mod.

Treba biti pažljiv prilikom sortiranja opsega koji sadrže formule. *Lotus* će izvršiti prilagođenje relativnih koordinata, međutim, ako se formule pozivaju na ćelije iz opsega koji se sortira, prilagođenje neće biti korektno. Neće biti problema sa formulama koje se pozivaju na ćelije izvan opsega, kao ni sa onim koje referišu ćelije unutar istog rekorda (reda).

/Data Query

U ovu grupu smještene su komande za rad sa bazom podataka.

Input, Criterion i **Output** opcije koriste se za zadavanje specifičnih opsega za rad sa bazom podataka.

Ulazni opseg **(Input Range)** predstavlja ertavu bazu podataka. Obuhvata imena polja i sve kartona.

Opseg kriterija **(Criterion Range)** označava opseg gdje će program tražiti uslove za pretraživanje baze. Obično se formira ispod ili pored baze. Sadrži red sa imenima polja (maksimalno 32), kao i nekoliko redova ispod u kojima se zadaju uslovi. Pošto nazivi polja moraju biti isti kao i u bazi, najbolje ih je iskopirati

bojom ili šrafurom. Opseg A je uvijek posljednji, a na njega se slažu ostali opsezi. X opseg se koristi da označi labele se tačke uzduž horizontalne ose. Numerička skala na Y osi se automatski prikazuje.

U slučaju kružnog ili torta, dijagrama **(Pie Chart)**, sve vrijednosti opsega se tretiraju kao dijelovi jedne cjeline. Pri tome obično nisu bitne same numeričke vrijednosti, već njihov međusobni odnos i odnos prema cjelini. Podaci se naznačavaju A opsegom. Svaki isječak kružnog dijagrama može biti prikazan drugom bojom ili šrafurom. Za označavanje boje (šrafure) koriste se numerički kodovi od 1 do 7. Kodovi 0 i 8 označavaju isječke bez boje (šrafure). Ako se nekoj vrijednosti doda 100, taj isječak će biti naglašen izvlačenjem iz kruga. Opseg B se može iskoristiti za označavanje seta podataka o bojama. To može biti bilo koja prazna oblast na tabeli istih dimenzija kao i A opseg. X opseg se koristi za označavanje seta labela koje će biti prikazane uz svaki isječak. Uz to, automatski se izračunavaju i prikazuju procenti pojedinih isječaka.

/Graph X A B C D E F

Omogućava da se po jednom opsegu pridruži set vrijednosti iz tabele.

/Graph Reset

Poništava definisane parametre ili opsege.

/Graph Save

Pomoću ove komande možemo tekući graf snimiti na disk. Datoteka automatski dobija nastavak PIC. Ta datoteka se koristi kao ulaz u PrintGraph program i ne može se koristiti za naknadno učitavanje i modifikaciju grafa na ekranu.

/Graph Options

U ovu grupu smještene su komande za postavljanje mnogobrojnih parametara za prikaz dijagrama.

Opcija **Legend** omogućava pravljenje tzv. legende, tako da se zna šta na grafu predstavljaju koja boja ili šrafura.

Opcijom **Format** kontroliše se prikaz linije za linijski i XY graf.

Titles opcijom mogu se naznačiti naslovi za obe ose i cijeli graf.

Opcijom **Grid** dodajemo ili uklanjamo mrežu sa grafa.

Opcijom **Scale** možemo postaviti skalu za X i Y ose ili odrediti „faktor preskakanja" podataka u X opsegu.

Opcije **B&W** i **Color** omogućuju crno-bijeli ili prikaz u boji.

Data-Labels opcija omogućava „labeliranje" odgovarajućeg opsega na grafu. U slučaju linijskih ili XY grafova labele mogu biti prikazane centrirano, lijevo, desno, iznad ili ispod tačke odgovarajućeg podatka. Kod stubičnih dijagrama labele se prikazuju iznad pozitivnih i ispod negativnih stubaca. Kod kružnih dijagrama labele se ne prikazuju.

/Graph Name

Omogućava da se tekući graf i njegovi parametri sačuvaju pod nekim imenom, kako bi se mogli koristiti i kasnije. Ovim je, takođe, omogućeno da se tabelu kreiramo više grafova i da ih brzo preklapamo.

Opcijom **Create** dajemo ime grafu.

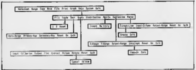
Opcijom **Use** biramo neki od imenovanih grafova.

Opcijom **Delete** brišemo neki od definisanih grafova.

/Graph Quit

Izborom ove komande vraćamo se u glavni meni.

/Data



Data komande koriste se za unos i analizu podataka u tabeli. Komande iz ove grupe su vrlo moćna i iskusniji korisnici će ih često koristiti. One, na primjer, omogućavaju da se na osnovu sekvence podataka formiraju tabele u kojima se može pratiti kako se mijenjaju vrijednosti formula u ovisnosti od ulaznih podataka.

Neke od komandi rade sa dijelovima tabela koji su organizovani kao baza podataka. To znači da su podaci smješteni u zapise. Svaki zapis je u stvari red u tabeli. Zapis sadrži određene informacije koje su

Postoje dva tipa tabelarnih **(Worksheet)** komandi. Jedne se odnose na kompletnu tabelu, dok druge imaju uticaja na dijelove tabele.

Prva grupa komandi naziva se globalnim i smještena je u **Global podmeniju**. Ove komande omogućavaju postavljanje opštih radnih parametara za tabelu, kao što su: format brojeva, poravnanje labela, širinu kolona, status zaštite, način rekalkulacije i slično. Takođe, ovom grupom komandi omogućeno je postavljanje raznih predefinisanih (default) vrijednosti (valuta, format za datum i vrijeme, radni direktorij, tip štampača itd.) koje se ne mijenjaju tako često. Tako se i komande iz ove grupe ne koriste često.

Druga grupa komandi koristi se kada je potrebno raditi samo sa dijelovima tabele, manipulisati se redovima i kolonama i postavljati nove lokalne parametre koji će imati uticaja samo na ograničenu grupu ćelija. Primjenom komandi iz ove grupe omogućeno je da se lokalno mijenjaju globalno postavljenih parametara. Ove komande takođe omogućuju dodavanje i brisanje redova i kolona.

/Worksheet Global Format

Ovom komandom se određuje u kom će se formatu pojavljivati brojevi na tabeli. Ako su otvorena dva prozora, ova komanda djeluje samo na aktivni prozor, tako da se isti dio tabele može vidjeti u dva različita formata istovremeno.

Na raspolaganju su sljedeći formati:

Fixed — broj sa fiksnim brojem decimala (0—15)
Scientific — broj u formi mantise i eksponenta
Currency
— hiljade, milioni itd. odvojeni zarezom, ispred broja stoji oznaka za valutu, negativni brojevi u zagradi
General — U zavisnosti šta je pogodnije broj se prikazuje u **Fixed**, ili **Scientific** formatu
,/, hiljade, milioni itd. odvojeni zarezom, negativni brojevi u zagradi
+/- — brojevi u vidu horizontalnih linija od znakova+(pozitivni) i —(negativni)
Percent — broj sa dvije decimale i znakom %
Date — neki od mogućih formata za datum i vrijeme
Text — formule prikazane u vidu teksta
Hidden — „sakriva ćelije" — ne pojavljuju se ni na ekranu ni na štampaču

/Worksheet Global Label-Prefix

Određuje koji će se prefiks podrazumijevati pri unosu labele (komentara) u tabelu, tj. da li će inicijalno labela biti poravnate po lijevoj ivici, centrirane, ili će biti poravnate po desnoj ivici ćelije.

/Worksheet Global Column-Widht

Postavlja širinu ćelija u tabeli. Podrazumijeva se vrijednost 9, ali se može varirati od 1 do 72. Širina je izražena u broju znakova koji mogu stati u ćeliju.

/Worksheet Global Recalculation

Ovom komandom određuje se način rakalkulacije.

Inicijalno je rakalkulacija postavljena na **Automatic**, tako da se prilikom promjene bilo kojeg podatka preračunava čitava tabela. **Lotus**, na žalost, nema „pametnu" rakalkulaciju koja bi preračunavala samo formule na koje promijenjeni podatak ima uticaja. Ako unosimo ili mijenjamo podatka u velikoj tabeli, preporučljivo je rekalkulaciju postaviti na **Manual**, jer bi u protivnom rad bio bitno usporen, pošto bi se nakon svakog unosa moralo sačekati na rakalkulaciju. U tom slučaju rakalkulacije se vrši nakon pritiska na funkcijsku tipku **F9**.

Osim ovoga, može se postaviti i način odvijanja proračuna, koji može biti prirodan **(Natural)**, po kolonama **(Columnwise)**, ili po redovima **(Rowwise)**. Izborom opcije **Iteration** postavlja se broj ciklusa rakalkulacije ako u tabeli postoje kružne reference.

/Worksheet Global Protection

Korištenjem ove komande omogućuje se kontrola pristupa pojedinim dijelovima tabele. Naime, možemo zamisliti da oko svake ćelije postoji „električna ograda". Kada nije puštena struja, ćelijama se može slobodno pristupati i mijenjati njihov sadržaj. Međutim, ako izborom opcije **Enable** uključimo struju, više nije moguće mijenjati sadržaj zaštićenih ćelija.

Sa pojedinih ćelija (komandom **/Range Unprotect)** može se potpuno skinuti „ograda", tako da one i nakon uključenja zaštite ostaju pristupačne. Na ovaj način se tabela može zaštititi od neželjenih slučajnih promjena, koje bi, na primjer, mogle nastati upisivanjem podataka u ćelije sa formulama. Zaštićene ćelije postaju ponovo pristupačne izborom opcije **Disable**.

/Worksheet Global Default

Koristi se za određivanje vrste štampača i interfejsa, kao i disk jedinice koja se inicijalno koristi za smještanje podataka.

/Worksheet Global Zero

Određuje da li će nule biti prikazane na tabeli ili ne.

/Worksheet Insert

Koristi se za ubacivanje praznih redova ili kolona u tabelu. Prazan red se ubacuje ispod reda, a kolona

desno od kolone na kojoj se nalazi pokazivač. Ukoliko je potrebno može se odjednom ubaciti i više redova ili kolona, tako što će se u komandi naznačiti opseg. Prilikom ubacivanja stari sadržaj tabele biće pomjeren nadole (redovi) ili udesno (kolone), a sve formule će automatski biti prilagođene, tako da referišu na iste ćelije kao i prije.

/Worksheet Delete

Briše redove ili kolone iz tabele. Inicijalno se briše red ili kolona na kojoj se nalazi pokazivač, ali se može naznačiti i opseg za brisanje. Brisanjem redova sadržaj ispod se pomiče na gore, a brisanjem kolona sadržaj se pomjera ulijevo. Ukoliko se briše opseg na koji se pozivaju neke formule, one će vratiti poruku ERR. Pri korištenju ove komande treba biti pažljiv jer se sadržaj obrisanih redova/kolona trajno gubi. Ako se radi samo o preuređenju tabela u smislu smanjenja udaljenosti pojedinih blokova, preporučljivije je koristiti komandu **/Move**.

Kada se brišu ćelije se podacima, oslobađa se dio memorije koji su zauzimale te ćelije, no da bi se oslobođena memorija potpuno iskoristila, potrebno je tabelu snimiti na disk, a zatim je ponovo učitati.

/Worksheet Column-Width

Mjenja širinu pojedinačne kolone.

Često se dešava da inicijalno postavljena širina ćelije nije dovoljna da prikaže pojedine brojeve u tabeli, pa se u tim ćelijama pojavljuje niz zvjezdica. U tom slučaju treba povećati širinu tih kolona. Takođe u tabelama se često ostavljaju prazne kolone koje razdvajaju grupe podataka, i koje je ponekad, zbog širine prikaza, potrebno smanjivati. Ove akcije se ostvaruju izborom opcije **Set**. Moguće je direktno zadati novu širinu kolona ili koristeći strjelice sa ljevo i desno izvršiti podešavanje u koracima. Pri tome su rezultati formatiranja odmah vidljivi.

Izborom opcije **Reset** širina kolone se vraća na globalnu vrijednost.

/Worksheet Erase

Briše kompletnu tabelu iz memorije. Ako podaci nisu prethodno snimljeni na disk, biće trajno izgubljeni. Ova komanda se ne koristi često, uglavnom kada nakon početnog skiciranja želimo početi iznova rad na praznoj tabeli. Za slučaj kada nakon završetka rada se jednom tabelom želimo nastaviti rad sa drugom, već postojećom tabelom, dovoljno je koristiti komandu **/File Retrieve**, koja automatski briše stari sadržaj i učitava novu tabelu.

/Worksheet Titles

Omogućava „zamrzavanje" specificiranih redova i kolona uz gornji ili lijevi rub ekrana, tako da su uvijek vidljivi, bez obzira na kretanje po tabeli. Obično se „zamrzavaju" zaglavlja redova i kolona. Izborom opcije **Horizontal**, svi redovi iznad pokazivača biće tretirani kao horizontalni naslov i na njih neće uticati vertikalno skrolovanje ekrana. Horizontalno skrolovanje odvijaće se kao i obično. Analogno tome, izborom opcije **Vertical**, sve kolone lijevo od pokazivača činiće vertikalni naslov i biće neosjetljive na horizontalno skrolovanje. Opcija **Both** omogućava istovremeno određivanje i horizontalnih i vertikalnih naslova.

Opcijom **Clear** „odmrzavamo" prethodno definisane naslove.

/Worksheet Window

Omogućava podjelu ekrana na dva horizontalna ili vertikalna prozora. Ekran se dijeli na poziciji pokazivača na način koji zavisi od izbora opcije **Horizontal**, ili **Vertical**. Ovo omogućava istovremeno posmatranje dva različita dijela tabele. Tako, na primjer, možemo vidjeti kako se promjene u bloku se ulaznim podacima reflektuje na blok se konačnim rezultatima. Mada se radi o jednoj tabeli, ovi prozori se ponašaju i kao posebne tabele jer se za svaki prozor mogu zadati drugačiji parametri formatiranja, na primjer, širina kolona, čak i ako pokazuju na istu oblast tabele. Iz prozora u prozor prebacujemo se uz pomoć funkcijske tipke F6. Prozori se izborom opcije **Sync**, mogu uskladiti (sinhronizovati), tako da skrolovanje u jednom izazva ekvivalentnu akciju u drugom prozoru. Pri tome se horizontalni prozori sinhronizuju na horizontalno, a vertikalni na vertikalno skrolovanje. Izborom opcije **Unsync** skrolovanje postaje ponovo nezavisno, a izborom opcije **Clear** vraćamo se na jedan prozor.

/Worksheet Status

Prikazuje raspoloživu memoriju i globalne parametre tabele.

/Worksheet Page

Ubacuje oznaku za početak nove strane u tabelu. U redu na kojem se nalazi pokazivač, u koloni A pojavljuje se znak :: (dvostruka dvotačka), a ostatak tabela se pomjera naniže. U taj red ne treba ništa upisivati, jer će kod ispisa takav sadržaj biti ignorisan. Da bi se oznaka prekida strane uklonila, može se obrisati komandom **/Range Erase**, ili, jednostavno, na njeno mjesto upisati novi podatak.

/Range

U radu se tabelom često je potrebno naznačiti opseg ćelija nad kojima će se izvršiti odgovarajuća akcija. To može biti označavanje zbog štampanja ili snimanja na disk, određivanje opsega pri radu sa bazom podataka, a često i funkcije zahtjevaju opseg kao argument. Zbog toga, kao i zbog niza drugih operacija, postoji čitava klasa komandi koje omogućavaju rad se opsezima ćelija. Opseg se može izabrati na jedan od tri načina

Vjerovatno će trebati malo eksperimentisanja dok se na izlazu ne dobiju željeni rezultati. Mogući problemi su:

Dio tabele se štampa normalno, a dio na kraju dokumenta. Razlog je što opseg ne može da stane između postavljenih margina (lijeve i desne), pa treba podesiti margine. Nekad je to i korisno, jer omogućava naknadno spajanje dijelova u kompletnu tabelu. Ako je desna margina veća od širine papira, dio linije koji ne može da stane biće štampan u novom redu. U tom slučaju treba pokušati štampati kondenzovanim karakterima. Ako to ne upali, onda treba smanjiti desnu marginu i dijelove odvojeno štampati.

I u samu tabelu mogu se ubaciti naredbe za kontrolu štampača. U ćeliju je potrebno upisati znak | (uspravna crta), a zatim kontrolne kodove razdvojene znacima \ (Obrnuta kosa crta). To omogućava da se na primjer dio tabele štampa normalnim, a dio kondenzovanim znakovima.

Na kraju, onima koji namjeravaju raditi se vrlo širokim tabelama, preporučuje se nabavka nekog programa *Sideways* tipa. Oni omogućavaju ispisivanje teksta sadranog u print (ASCII) datoteci po dužini papira.

/Print Printer/File Clear

Resetuje sve parametre na njihove predefinisane (default) vrijednosti.

/Print Printer/File Align

Saopštava printeru da je papir postavljen na početak strane. Ovo bi trebalo uraditi prije svakog štampanja tabele, jer bi se u protivnom mogle pojaviti „rupe" usred štampane strane.

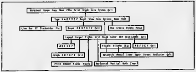
/Print Printer/File Go

Starta proces štampanja.

/Print Printer/File Quit

Kraj procesa štampanja. Neophodno ako je izlaz preusmjeren u datoteku.

/Graph



Lotus omogućava da se podaci u tabeli prikažu i grafički. Komanda svrstane u ove grupu namijenjene su upravo radu se grafovima. Na raspolaganju je nekoliko hpova dijagrama ili grafova.

Linijski graf daje jasan pregled promjene neke vrijednosti u vremenu. Stubični dijagrami koriste se da prikažu razlike među pojedinim podacima. XY grafom se može prikazati međusobna zavisnost dvije grupe podataka. Kružni (torta) dijagrami su namijenjeni za prikaz zastupljenosti dijelova u cjelini. Grafovi se automatski prilagođavaju prilikom promjena podataka.

Da bi se graf mogao vidjeti na ekranu, neophodno je imati grafičku video kartu. Međutim, odsustvo grafičke karte ne sprečava da se grafovi kreiraju i odštampaju na štampaču ili nacrtaju na ploteru.

/Graph Type

Omogućava izbor tipa grafa za prikaz podataka.

Pomoću linijskog grafa (Line Graph) sve vrijednosti u definisanom opsegu predstavljaju se kao tačke na odgovarajućoj udaljenosti iznad horizontalne ose. Tačke su povezane linijama. Može se kreirati do šest linija na istom grafu, što omogućuje uporedni prikaz raznih grupa podataka. Za podatke se koriste opsezi A-F, a X opseg se koristi da označi labele za tačke uzduž horizontalne (X) ose. Numerička skala na Y osi se automatski prikazuje.

Stubični dijagram (**Bar Graph**), prikazuje podatke u specificiranom opsegu kao stupce promjenjive dužina. Istovremeno je moguće prikazati do šest setova (opsega) podataka. Stupci svakog seta su prikazani drugom bojom ili šrafurom. Za podatke se koriste opsezi A-F, a X opseg se koristi za označi labele za tačke uzduž horizontalne (X) ose. Numerička skala na Y osi se automatski prikazuje.

XY graf omogućava uparivanje svake vrijednosti u X opsegu sa vrijednostima u opsezima **A-F**. Tačke svakog opsega označene su drugačijim simbolom. Numerička skala na X i Y osi se automatski prikazuje.

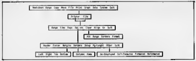
Slaganim stubičnim dijagramom (**Stacked-Bar Graph**) se sve vrijednosti jednog opsega prikazuju pomoću jednog stupca čija je visina jednaka zbiru visina pojedinačnih stubaca. Tako se može jasno prikazati kako koja vrijednost učestvuje u ukupnom zbiru. Sve vrijednosti opsega prikazana su drugačijom

/File Directory

Mijenja tekući direktorij.

Novi direktorij se zadaje tako da se otkuca njegovo ime i put do njega. Put može početi i iz tekućeg direktorija ako je novi direktorij u nekom podnivou tekućeg direktorija. Sve buduće operacije sa diskom podrazumijevaće taj direkorij kao radni.

/Print



U ovu grupu svrstane su komande kojima se tabela ili neki njeni dijelovi mogu ispisati na štampaču ili poslati u print datoteku. Sve komande su ravnopravne za oba slučaja s tim što, ako izaberemo opciju **File**, trebamo navesti na već uobičajeni način ime datoteke u koji ćemo preusmjeriti izlaz. Datoteka automatski dobija nastavak PRN.

Preusmjeravanje izlaza u datoteku omogućava da se sadržaj kasnije odštampa iz DOS-a, ili da se preuzme u neki drugi program, na primjer u procesor teksta, i dodatno obradi. Kontrola samog štampača je dobra. Moguće je pomjerati papir za liniju ili stranicu, postavljati margine, dužinu stranice, heder i futer itd. Naravno, prije nego što nešto ispišemo na štampaču, neophodno je u procesu instaliranja naznačiti koji štampač koristimo. Dobra je navika prije štampanja snimiti tabelu na disk.

/Print Printer/File Range

Uz pomoć ove komande označavamo opseg za štampanje. Pri tome treba provjeriti da li su označene sve potrebne informacije. Opseg se, kao i obično, može označiti pokazivačem, koordinatama ili imenom. Program pamti zadnji korišteni opseg, tako da ga kod ponovnog štampanja ne trebe naznačavati.

/Print Printer/File Page

Postavlja glavu štampača na kraj strane, ispisuje futer, ako je zadan, i prebacuje se na početak slijedeće strane.

/Print Printer/File Options

Komanda sadrži više opcija i omogućava postavljanje svih važnijih parametara za štampanje.

Izborom opcije **Header**, možemo zadati tekst koji će se pojaviti kao zaglavlje na početku svake stranice. Analogno, opcija **Footer** ostavlja mogućnost zadavanja teksta koji će se ponavljati na kraju svake stranice. Tekst se zadaje u jednoj liniji dužine do 240 znakova. Koliko će se znakova koristiti, zavisi od širine papira u štampaču. Ubacivanjem znaka # (hes, taraba) mogu se automatski generisati brojevi strana, a ako ubacimo znak @ (et, ludo a) dobićemo tekući datum. Heder i futer mogu imati do tri sekcije. Prva je lijevo poravnata, druga je centrirana, a treća poravnata po desnoj ivici. Za razdvajanje sekcija koristi se znak | (uspravna crta).

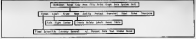
Opcije **Margins** omogućava postavljanje margine. Pri tome su prikazane tekuće vrijednosti. Za promjenu, dovoljno je ukucati nove vrijednost.

Opcijom **Borders** imamo mogućnost da specificiramo redove ili kolone iz tabele, koji će biti štampani na svakoj strani lijevo ili iznad opsega koji smo naznačili za štampanje. Ovo je u neku ruku slično „zamrzavanju" redova, ili kolona na ekranu. Redove i kolone naznačene na ovaj način ne treba uključivati u opseg za štampanje jer bi se dobile dvije kopije istih.

Setup opcija ostavlja mogućnost da se štampaču pošalje niz kontrolnih znakova. Niz može bit dug maksimalno 39 znakova. Sastoji se obično od trocifrenih kodova razdvojenih znakova \ (beksles, obrnuta kosa crta). Koji će to kodovi biti zavisi od samog štampača i od toga šta želimo da postignemo.

Pomoću opcije **Page-Lenght** određujemo dužinu stranice u linijama. U dužinu se računaju i gornje i donje margina, heder i futer, kao i po dvije prazne linije ispod hedera i iznad futera, koje Lotus automatski ubacuje.

Iz **Options** podmenije možemo izabrati više opcija. **As-Displayed** ispisuje opseg onako kako je prikazan na ekranu. **Cell-Fomaulas** omogućuje ispis svake pune ćelije u opsegu, tako da sadržaj ćelije zauzima jednu liniju u ispisu. Izborom opcija **Formated** i **Unformated** može se birati između formatiranog i neformatiranog ispisa opsega.



— označavanjem renga pokazivačem (rang će biti prikazan inverzno)
— direktnim unosom koordinata ćelija dva suprotna ugla
— kucanjem imena opsega, ili njegovim izborom iz menija

U nastavku će biti dati kraći opisi komandi iz ove kategorije.

/Range Format

Komande iz ove grupe služe za lokalnu promjenu globalno postavljenih elemenata formata tabele. Opcije i rezultati formatiranja su ekvivalentne onima iz **/Worksheet Global Format** menija, samo što se format odnosi na označeni opseg ćelija. Pri tome treba imati u vidu da promjena formata prikaza nema utjecaja na vrijednost brojeva u ćelijama. Ona ostaju nepromjenjene.

Tako ako je broj imao decimalnu vrijednost, izborom **Fixed** formata bez decimalnih mjesta u ćeliji će biti prikazan broj zaokružen na najbližu cijelu vrijednost, ali će vrijednost broja ostati nepromijenjena i kao takva će se koristiti u računanju.

Izborom opcije **Hidden** mogu se pojedini dijelovi tabela učiniti nevidljivim. Opcija **Reset** vraća format u opsegu na globalne vrijednosti.

/Range Label

Određuje način poravnanja postojećih labela u naznačenom opsegu. Na može se odrediti način poravnanja prije nego što se unesu labela.

/Range Erase

Briše sadržaj ćelija u navedenom opsegu. Brisanjem sadržaja, međutim, ne brišu se informacije o numeričkom formatu i statusu zaštite. Takođe, ne mogu se obrisati zaštićene ćelije dok je zaštita aktivna.

/Range Name

U ovu grupu su smještene vrlo interesantne i korisne komande. Ukratko, one omogućavaju da se umjesto koordinatama u referisanju ćelije u tabeli koristimo imenima. Modeliranje tako postaje bitno lakše i razumljivije.

Opcije **Create** omogućava da se često korištenom opsegu da ime dugo do 14 karaktera. Zbog mogućih zabuna u imenima opsega, treba izbjegavati znakove matematičkih operacije ili razmake. Ne pravi se razlika između malih i velikih slova. Istom opcijom moguće je promijeniti definiciju već imenovanih opsega ili samo pregledati na šta se oni odnose. To se postiže tako što umjesto da otkucamo novo ime izaberemo odgovarajuće iz ponuđene liste. Kada je opseg jednom imenovan, može se koristiti u svim formulama gdje je to potrebno, ne vodeći računa o stvarnim koordinatama ćelija. Definicije i imena opsega čuvaju se zajedno sa tabelom.

Opcijom **Delete** briše se ime opsega. Time se ne mijenja sadržaj ćelije u opsegu, već samo briše ime iz liste.

Opcijom **Labels** na brz način se mogu pojedinačnim numeričkim ćelijama dodijeliti imena. Ta imena su, u stvari, labela koje se nalaze neposredno uz ćelije sa podacima. Tim imenima se služimo kao varijablama u pisanju formula. Prilikom izbora opsega određuje se i sa koje strane labela se nalaze podaci. Ako su tabele već korištene, nove vrijednosti će se koristiti umjesto starih, ali će i stare formule biti valjane, samo što će ćelije ponovo referisati na osnovu koordinata, a ne imena.

Opcija **Reset** briše sva definisana imena za opsege. Time se sadržaji opsega ne mijenjaju, a i sve formule ostaju valjane.

Opcije **Table** daje alfabetski sortiranu listu svih definisanih opsega. Lista se sastoji od dvije kolone. U jednoj su imena, a u drugoj koordinate. Pri izboru ove opcije treba odrediti opseg za ispis liste. Pri tome treba paziti da lista ne prepiše neke podatke na tabeli.

/Range Justify

Tretira kolonu sa tekstom kao paragraf. Tekst formatira tako da nijedna linije nije duža od zadane vrijednosti. Ovu komandu ne treba koristiti na opseg u koma ima zaštićenih ćelija.

/Range Protect

Koristi se za postavljanje zaštite na opseg ćelija. Zaštita nije u funkciji sve dok se ne aktivira komandom **/Worksheet Global Protection Enable**

/Range Unprotest

Skida zaštitnu ogradu sa opsega ćelija. One i nakon aktiviranja zaštite ostaju slobodne za promjene.

/Range Input

Korištenjem ove komande može se ograničiti kretanje pokazivača samo na nezaštićene ćelije. Ovo pruža mnoge pogodnosti u izradi sopstvenih aplikacija, pogotovo onih koje će drugi koristiti. Dobri rezultati se postižu u kombinaciji sa makroima.

/Range Value

Omogućava konvertovanje formula u opsegu u njihove vrijednosti. Nakon izbora komande potrebno je odrediti dva opsega — izvorni (FROM) i odredišni (TO). Ukoliko su izvorni i odredišni opsezi jednaki, ćelije koje su u opsegu sadržale formule sadržaće samo njihove vrijednosti. Ukoliko se opsezi razlikuju na odredište će biti iskopirane vrijednosti formula ali ne i same formule. Izvorne formule neće biti promijenjene. Za kopiranje formula koristi se kimanda /Copy. Primjenom ove komande može se uštedjeti nešto memorije. Da bi se uštede primijetila potrebno je tabelu snimiti, a zatim je ponovo učitati sa diska.

/Range Transpose

Vrši kopiranje opsega tako da u odredišnom opsegu redovi postaju kolone, a kolone redovi. Za odredišni opseg dovoljno je zadati samo gornji lijevi ugao. Komanda ne radi ako se koordinate lijevog ugla izvorišnog i odredišnog opsega poklapaju. Ovu komandu ne treba primjenjivati na opsege koji sadrže formule sa relativnim koordinatama jer prilikom kopiranja neće doći do prilagođenja koordinata.

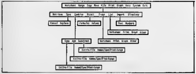
/Copy

Koristi se za kopiranje opsega. Prilikom kopiranja opsega sa formulama dolazi do prilagođenja svih relativnih koordinata. Apsolutne koordinate ostaju nepromijenjene. Komanda zahtijeva da se naznače dva opsega. Odgovarajućim zadavanjem čoredišnog opsega moguće je napraviti i višestruke kopije izvornog opsega. To se na primjer koristi kada je potrebno podvući crte u tabeli. Dovoljno je u jednu ćeliju u redu u kom ćemo podvući crtu upisati formulu \ — . Time će znak – ispuniti ćeliju sa formulom. Potrebno je zatim tu ćeliju označiti kao izvorni opseg, a kao odredišni opseg naznačiti ostale ćelije u redu i u čitavom redu će se pojaviti linija.

/Move

Premješta naznačeni opseg na novu lokaciju u tabeli. Pri tome, ako se premješta ćelija sa formulom, ona ostaje ista, a ako se premješta ćelija na koju se poziva formula, formula se prilagođava tako da rezultat ostaje isti. Ako se pomjera ćelija u pornjem lijevom ili donjem desnom uglu nekog imenovanog opsega, mijenja se i sama definicija opsega. Komanda je jako dobro izvedena i bez ikakvog straha da će se pokvariti rezultati može se vršiti pomjeranje dijelova tabele.

/File



U ovom meniju sakupljene su komande za rad sa diskom. Uz pomoć njih može se tabela ili njeni dijelovi snimiti na disk ili se sa njega pročitati kombinujući pri tome dijelove raznih tabela. Ostale komande omogućuju brisanje datoteka sa diska i import ASCII datoteka u tabelu. Pri radu sa datotekama u vezi sa njihovim imenima, važe sva pravila kao i u DOS-u. Osim direktnim navođenjem imena datoteka se mogu birati i iz liste. Nazivi u listi koji nakon imena imaju znak \ (obrnuta kosa crta, backslash) predstavljaju direktorija.

/File Retrieve

Učitava tabelu sa diska u memoriju računara i prikazuje je na ekranu. Učitavanjem nove tabele biće pobrisan prethodni sadržaj memorija. Zato tekuću tabelu prije početka rada sa novom treba snimiti na disk. Ime datoteka se može direktno otkucati ili se može izabrati iz prikazane liste. Dok je prikazana lista raspoloživih datoteka, pritiskom na funkcijsku tipku F3 (NAME) mogu se vidjeti kompletne informacije o tekućoj datoteci — datum i vrijeme zadnje promjene i dužina datoteke. Ponovnim pritiskom na F3 vraćamo se u /File meni. Ukoliko želimo učitati datoteku iz nekog drugog direktorija, potrebno je prije imena datoteke navesti kompletan put polazeći od glavnog direktorija. Ako je tabela snimljena sa zaštitom, zahtijevaće se unos odgovarajuće lozinke. Promjene koje se vrše u tabeli nemaju uticaja na datoteku na disku, sve dok se tabela ne snimi. Zato u slučaju da nešto zabrljamo ili se predomislimo, možemo jednostavno tabelu ponovo učitati i početi rad iznova.

/File Save

Snima tekuću tabelu na disk. Uz tabelu čuvaju se i svi parametri koji su postavljeni u toku rada. Da bi se tabela snimila potrebno je otkucati ime datoteke ili prihvatiti već ponuđeno. Ostavljena je i mogućnost da se predomislimo. Ako izaberemo opciju **Cancel** vraćamo se i READY mod, a tabela nije sačuvana. Opcijom **Replace** azuriramo već postojeću datoteku.

Svoje tabele možemo zaštititi od neovlašćenog korištenja zadavanjem lozinke (password). Procedura se razlikuje od standardne utoliko što nakon unosa imena datoteke treba pritisnuti razmaknicu (SPACE) i otkucati P, a zatim unijeti i odgovarajuću lozinku. Lozinka može biti duga do 15 karaktera i pošto nije vidljiva na ekranu, potrebno ju je još jednom otkucati zbog verifikacije. Pri ovome treba biti pažljiv jer ako zaboravimo lozinku, nećemo više moći pročitati tabelu.

Ako tabeli se kojom stalno radimo damo ime AUTO123, ona će se automatski učitavati svaki put kada pokrenemo program.

/File Combine

Omogućava da se tekuća tabela kombinuje sa drugim tabelama ili njihovim dijelovima. Pri tome se pokazivač koristi kao oznaka gornjeg lijevog ugla oblasti u koju će se ubaciti novi podaci. Podaci se ubacuju desno i ispod pokazivača. U tekuću tabelu ne prenose se parametri formatiranja.

Mogu se izabrati tri načina kombinovanja podataka u rednoj tabeli i novih podataka. Opcijom **Copy** stare podatke zamjenjujemo novim. Ako je izabrana opcija Add, novi podaci se sabiraju sa već postojećim, a ako je izabrana opcija **Subtract**, u ćelijama naznačene oblasti pojaviće se razlika starih i novih podataka. Oduzimanjem pozitivne vrijednosti od prazne ćelije dobija se negativna vrijednost.

Nakon izbora načina kombinovanja, potrebno je odrediti i šta se kombinuje. Ako izaberemo **Entire-File**, uzima se čitava tabela, u protivnom, ako izaberemo **Named/Specified-Range**, uzima se imenovani ili koordinatama specificirani opseg iz tabele. Tabela se bira bilo navođenjem imena bilo izborom iz liste.

/File Xtract

Omogućuje da se dio tekuće tabele izdvoji i sačuva u posebnoj datoteci. To ne izaziva promjene u radnoj tabeli. Izdvojeni dio može se koristiti kasnije kao nezavisna tabela, jer će osim podataka biti sačuvani parametri formatiranja, kao i ostali parametri (opsezi, grafovi itd.).

Ako se prilikom izdvajanja izabere opcija **Formulas**, u novu datoteku formule će biti prenesene onakve kakve jesu. Pri tome treba voditi računa da formula ili opsezi, ne referišu ćelije izvan opsega za izdvajanje. Opcijom **Values** umjesto formula, u novu datoteku biće prenešene njihove vrijednosti.

/File Erase

Korišćenjem ove komande mogu se sa diska obrisati datoteke određenog tipa.

Izborom opcije **Worksheet** dobija se lista svih datoteka u tekućem direktoriju sa nastavcima WK1 i WKS. Analogno, opcija **Print** daje listu datoteka sa nastavkom PRN, a opcije **Graph** listu datoteka sa nastavkom PIC. Opcija **Other** daje listu svih datoteka.

Datoteka za brisanje može se izabrati iz liste ili se direktno može otkucati njeno ime. Pri tome je dozvoljena upotreba džoker karaktera (* i ?) kao i u DOS-u. Time se brišu sve datoteke koje zadovoljavaju dati uslov.

/File List

Prikazuje sve datoteke određenog tipa u tekućem direktoriju, kao i preostali prostor na disku.

Sa ekrana privremeno nestaje tabela, a zamjenjuje je lista datoteka. Po listi se možemo kretati koristeći kurzorske strjelice. Prikazane su i detaljnija informacija o trenutno osvijetljenoj datoteci. Ako nema datoteka specificiranog tipa, program omogućava da se zada novi uslov.

/File Import

Kopira print datoteku (nastavak PRN) u tabelu od pozicije pokazivača. Mogu se uvoziti datoteke iz svih programa koji prave ASCII datoteke na izlazu. Ako je nastavak drugačiji, treba ga promijeniti. Kontrolne karaktere u tekstu treba izbjegavati, jer mogu izazvati nepredvidive rezultate.

Opcijom **Text** sadržaj datoteka prevodimo u lijevo poravnate labele. Ako je izabrana opcija **Numbers**, iz datoteke se uzimaju samo brojevi koji postaju numeričke ćelije i tekst zatvoren u navodnicima čo kojih nastaju ćelije sa labelama.

Ograničenje na print datoteke je da tekst ne može biti širi od 240 i duži od 8192 karaktera.

Commodore Commodore Commodore Commodore

AMIGA 500

— Računar godine 1987.

U organizaciji stručne publikacije CHIP, međunarodni žiri iz 8 zemalja izabrao je mikroračunar godine za 1987. U kategoriji kućnih računara sa velikom prednošću pobedio je računar AMIGA 500.
Ocene stručnjaka su jednoglasne u tome da AMIGA 500, kao 16/32 bitni multitasking računar nove generacije i pravi naslednik računara C 64, postavlja nove standarde u toj kategoriji računara za 90-te godine.

Zastupnik **KONIM**
61000 Ljubljana, Titova 38
Telefon: 061/312-290

Prodajna mesta
Beograd „Metalservis"
Karađorđeva 65
Tel 011/624-927

Zagreb „Poljoopskrba"
Varšavska 13
Tel 041/428-796



AMIGA 2000

U osnovnoj varijati performanse računara su slične performansama AMIGE 500. Kao otvoren sistem daje mogućnost modularne dogradnje računara po željama kupaca:
- dodatna ugradnja 1×3,5 inča disketne jedinice i 1 hard disk, ili
- dodatna ugradnja 1×5,25 inča disketne jedinice i 1 hard disk
- sa ugradnjom dodatne kartice postižemo IBM-XT ili AT kompatibilnost
- 5 AMIGA dodatnih mesta za proširenja
- 4 PC XT/AT mesta za proširenja
- proširenje RAM do maksimalno 8,5 MB

Cena*
Osnovna izvedba $1 616.04

*** Na cenu u devizama dodati 60% dinarskih troškova**

Tehničke karakteristike	AMIGA 500
Mikroprocesor	MC 68000
Clock (MHz)	7 159
Dodatni procesori	8086, 4,77 MHz 80286, 8 MHz 68020
Tastatura	profesionalna sa 95 tastera (10 funkcijskih)
ROM	256 K
RAM	512 K maksimalno proširljivo na 8 5 MB
Grafika	320×256 (32 boje) 640×256 (16 boja) 320×512 (32 boje) 640×512 (16 boja) 320×256 (4096 nijansi)
Paleta	4096 boja
Ugrađena disketna jed.	3 5 inča 880 K
Dodatne disketne jedinice	3
Dodatni hard disk	20—80 MB
Interfejsi	IBM PC kompatibilni RS 232 i Centronics — analogni RGB video ulaz i izlaz — stereo audio ulaz i izlaz, UHF izlaz, 2 palice
Ostala proširenja	AMIGA sistemski bus (vodilo)
Operativni sistem	AMIGA DOS MS DOS (opcija)
Dobavljeni softver	AMIGA DOS Basic Workbench, CLI
Cene:	
AMIGA 500	$676 90
Color monitor 1084	$330 19
Printer MPS 1200P	$269 11
Disketna jedinica A 1010	$247,64

Predstavništva / Aussenstellen / Branch offices:
11070 BEOGRAD, Treći bulevar 120 E, Telefon (011) 138-556, Telex 11-498
71000 SARAJEVO, Boriše Kovačevića 14, Telefon (071) 24-503, Telex 41-215 yu jtx s
41000 ZAGREB, Svačićev trg 14, Telefon (041) 448-650, Telex 21-535 Z
91000 SKOPJE, Dame Gruev blok 2 lokal 4, Telefon (091) 238-463, Telex 51-878

C E N I K Št 06/88
od 01/04/88 do 30/04/88

		USA DOL	DIN
Računari			
1	PC 1	676 90	652 409 –
4	C–64	222 87	214 808 –
6	AMIGA 500	676 90	652 409 –
7	AMIGA 2000	1816 04*	1.750 333 –
9	PC 10/III	1139 15	1.097 934 –
10	PC 20/III	1642 69	1,583 256 –
11	C 128D	643 86	620 565 –
12	PC 40/20 (PC AT)	2517 69*	2 426 597 –
14	C–128	313 67	302 321 –
Periferijske jedinice			
20	KASETOFON 1530	29 71	28 635 –
30	DISKETNA JEDINICA 1541	236 09	227 548 –
31	DISKETNA JEDINICA 1570	264 15	254 592 –
33	DISKETNA JEDINICA 1571	297 17	286 419 –
34	DISKETNA JEDINICA A 1010	247 64*	239 680 –
40	MONITOR 1802	255 89	246 632 –
42	PC–MONITOR	123 82	119 340 –
43	MONITOR 1084	330 19*	318 244 –
62	ŠTAMPAČ MPS 1200	269 11	259 373 –
Oprema			
90	JOYSTICK 1311	5 79	5 795 –
92	JOYSTICK 1342	6 60	6 605 –
112	TRAK MPS 803	8 67	8 159 –
133	MODULI	6 60	6 605 –
139	MODULI	9 90	9 908 –
198	PROŠIRENJE RAM-a 2 MB 2 MB A2052	445 75	448 142 –
199	PROŠIRENJE 512 K ZA AMIGU	151 90	152 033 –
Proširenja			
200	256—512 K	99 06	104 654 –
205	HARD DISK 20 Mb	906 01	959 283 –
206	PROŠIRENJA 512—640 K	49 53	49 573 –
220	INT DISKETNA JEDINICA A2010	216 27*	216 460 –
235	PCAT TASTATURA	115 00	115 101 –
276	**MODULATOR A–520**	29 71	29 736 –

* Označene cene su snižene

Klub programera

Z80/spektrum

Vremenske petlje

Generisanje vremenskih intervala (pauza) može se javiti kao potreba, bilo da radimo sa mehaničkim periferijama (prekidačima i slično) koje zahtevaju neko vreme da se stabilizuju u svom novom položaju, bilo da proizvodimo pravilne periodične signale, recimo za generator tona.

Ovakvi problemi mogu se rešavati i *softverski* koristeći činjenicu da svaka instrukcija mikroprocesora Z80 traje tačno određen broj otkucaja kloka. Poznavajući frekvenciju kloka, možemo izračunati vreme izvršavanja bilo kog programa. Jasno, sve to pod uslovom da je klok *stalan*. Na „spektrumu" to znači da se program ne sme nalaziti u prvih 16K RAM-a, niti sme koristiti naredbe koje adresiraju taj deo memorije.

Najjednostavnija rutina za kašnjenje je obična *prazna petlja*, koja ništa drugo ne radi osim što troši vreme. Jedan prost primer bi bio

```
        LD      B,nn
WAIT    DJNZ    WAIT
```

Naredba **DJNZ** prvo umanjuje registar B za jedinicu, pa tek onda testira njegov sadržaj. Ako nije nula, vrši se skok i u tom slučaju izvršenje čitave naredbe traje 13 taktova. U protivnom prelazi se na sledeću instrukciju programa, a izvršenje traje samo 8 taktova. Koliko dugo će se petlja vrteti zavisi, naravno, od početnog sadržaja registra B. Ukupan broj taktova je

$n=B\times13-5$

Oduzeli smo 5 jer u poslednjem prolazu izvršenje ne traje 13 taktova, već osam.

Najduži interval je pri početnoj vrednosti B=0, i tada je $n=256\times13-5=3323$. Pri kloku od 3.4MHz (tipično za „spektrum") to iznosi vrlo blizu jedne milisekunde (0.98 ms).

Duži vremenski period možemo dobiti dodajući i telo petlje

```
        LD      B,nn
WAIT    neki niz naredbi
        DJNZ    WAIT
```

Ako pomenuti niz naredbi traje **N** taktova, onda cela petlja traje

$n=B\times(N+13)-5$

Izuzetno duge intervale (reda sekunda) na „spektrumu" najjednostavnije dobijamo koristeći instrukciju **HALT**, koja čeka sledeći *interapt* (a interapti se događaju 50 puta u sekundi, tj. svakih 20 ms). Recimo ovako:

```
        LD      B,nn
WAIT    HALT
        DJNZ    WAIT
```

U ovom slučaju trajanje same **DJNZ** naredbe može da se zanemari, pa je ukupno kašnjenje $n=B\times20ms$. Pri B=50 imamo tačno jednu sekundu.

U radu sa nekim periferijama potrebno je generisati intervale koji traju određen broj milisekundi. Potprogram koji obavlja takav posao mogao bi imati oblik

```
WAIT    LD      C,ONE_MS    (7)
WAIT_1  DEC     C           (4)
        JR      NZ,WAIT_1   (7/12)
        DEC     A           (4)
        JR      NZ,WAIT     (7/12)
        RET
```

U zagradi iza svake instrukcije dat je odgovarajući broj taktova radi lakšeg računa.

Konstanta DNE_MS zavisi od frekvencije kloka, a mi ćemo izračunati njenu vrednost za 3.4 MHz.

Broj milisekundi za kašnjenje nalazi se u akumulatoru pri pozivu potprograma. Unutrašnja petlja WAIT_1 treba da generiše osnovno kašnjenje od jedne milisekunde. Pri svakom prolazu ona troši 16 taktova, osim u poslednjem, kada traje 11 taktova. U zavisnosti od početnog sadržaja registra C (tj. od konstante DNE_MS) ukupno trajanje iznosi

$n_1=C\times16-5$

Slično tome, spoljašnja petlja WAIT izvršava se zavisno od početnog sadržaja A za

$n=A\times(7+n_1+16)-5$

gde je $7+n_1+16$ broj taktova pri jednom prolazu kroz WAIT petlju. Treba obezbediti da to iznosi tačno 1 ms:

$7+n_1+16=3.4\times10^6/1000$

Nije teško odatle izračunati da n_1 mora biti 3377, a početna vrednost registra C na ulazu u WAIT_1

```
ONE_MS  EQU     211
```

Time je sav račun završen. Ako sada, na primer, želimo da u programu negde sačekamo 7 ms, izvršićemo

```
        LD      A,7
        CALL    WAIT
```

Naravno, naredbe **CALL** i **RET** produžavaju malo željeni vremenski interval, ali to se može zanemariti.

Jovan Skuljan

Čitanje teksta

Naš saradnik Ranko Tomić iz Gornjeg Milanovca poslao nam je još jedan prilog za Klub programera. To je, kako kaže, kratka ali korisna rutina, koja će dobro doći igračima avantura, ali i svim ostalim koji vole da brljaju po tuđim programima.

Rutina služi za pregledanje teksta u memoriji, što se može iskoristiti u slučaju kada nabavite neku novu avanturu, a ne znate sve reči koje poznaje. Dovoljno je da rutinu asemblirate na neko prazno mesto u memoriji, učitate program koji želite da pregledate i startujete rutinu. Sav tekst koji se nalazi u memoriji biće ispisan na ekran ili, ako promenite broj u liniji 30, na štampač. Ispisuju se samo karakteri od 32 do 164, što se može promeniti ako vam ne odgovara (linije 280 i 300). Ako želite da malo bolje pogledate tekst, pritisnite 9 za pauzu. Pauza se isključuje na 0. Za prekid listanja dovoljno je pritisnuti SPACE.

```
10 HACK-1
20          ORG     64000       ;ili neka druga adresa
30          LD      A,2         ;upišite 3 za ispis na
                                 štampač
40          CALL    #1601
50          LD      HL,25000    ;adresa od koje počinje
                                 listanje
60 PETLJA
70          LD      A,255       ;sprečavanje poruke
                                 scroll?
80          LD      (23692),A
90          LD      A,239
100         IN      A,(254)
110         RRA
120         JR      NC,CONT     ;ako je pritisnuta 0 prekid
                                 pauze
130         RRA
140         JR      C,CHECK     ;ako nije pritisnuto 9 pro-
                                 veri da li je pauza
150         LD      A,1         ;inače počinje pauzu
160         OUT     (254),A
170         LD      (FLAG),A
180 CHECK
190         LD      A,(FLAG)
200         OR      A
210         JR      NZ,PETLJA   ;ako je pauza, ponovi pro-
                                 veru tastera
220 CONT
230         XOR     A           ;isključi pauzu
240         LD      (FLAG),A
250         LD      A,7
260         OUT     (254),A
270         LD      A,(HL)      ;proveri da li je bajt
                                 karakter
280         CP      32
290         JR      C,NEXT
300         CP      164
310         JR      NC,NEXT
320         RST     16          ;ako jeste, ispiše ga
330 NEXT
340         LD      A,127       ;proveri da li je pritisnut
                                 BREAK
350         IN      A,(254)
360         RRA
370         JR      NC,END      ;ako jeste, kraj
380         INC     HL
390         LD      A,H
400         OR      L
410         JR      NZ,PETLJA   ;ako nije kraj memorije po-
                                 novi ciklus
420 END
430         RET
440 FLAG    DEFB    0
```

Klub Z80/amstrad

Tokenizacija i ostvarivanje komandi

Često je potrebno omogućiti da računar prepozna neki tekst koji smo uneli sa tastature tako da u zavisnosti od otkucane reči preuzme određenu akciju. Takvo zadavanje komandi je, uprkos sve više (zlo)upotrebljavanim menijima, često najefikasnije i najpraktičnije rešenje. Meniji zauzimaju veliku površinu na ekranu i zahtevaju duže mašinske rutine za realizaciju. Korisniku koji često upotrebljava neki program je lakše da nauči komande nego da stalno šeta kroz nekoliko menija da bi ostvario šta želi. Sem toga, logično je da se programom koji zahteva tekstualni unos tekstom i upravlja, a ne pomoću miša ili, ne daj bože, džojstika.

Drugi put je pak potrebno da se sve rezervisane reči u nekom tekstu zamene jednim ili parom bajtova. To radi skoro svaki bejzik interpretator i veliki broj editora za druge programske jezike na kućnim računarima. Tom takozvanom tokenizacijom se sažima tekst i omogućuje da daleko duži programi stanu u memoriju i ubrzava njihovo kasnije prevođenje, odnosno izvršavanje.

Rutina u prvom listingu nalazi u tabeli reč koje odgovara unetom tekstu i vraća njen redni broj u akumulatoru. Tabela se sastoji od reči napisanih velikim slovima, a poslednji karakter u svakoj reči je uvećan za 128 da bi se setovanim najvišim bitom označio kraj te i početak sledeće reči. Rutina prepoznaje reči unete i velikim i malim slovima. Ulazni parametri su sme-

LDIR preporučeno

Saše Pušica, 9. brigade 17/2, Bor. Rutina za štampanje teksta slovima dvostruke visine koju ste nam poslali jeste interesantna, ali o tome smo u „Računarima" već pisali. Setite se, recimo, teksta „Operacija WRITE" iz „Računara 16".

```
Listing 1
TOKEN    PUSH   BC
         LD     B 0
MAIN     PUSH   HL
T_LOOP   LD     A (DE)
         LD     C (HL)
         RES    5 C
         BIT    7,A
         JR     NZ LAST
         CP     C
         JR     NZ NEXT
         INC    HL
         INC    DE
         JR     T_LOOP
LAST     AND    7F
         CP     C
         JR     NZ NEXT
         INC    HL
         LD     A (HL)
         CALL   TEST
         JR     C NEXT
         LD     A B
         POP    BC
         POP    BC
         SCF
         RET
NEXT     EX     DE HL
N_LOOP   BIT    7,(HL)
         INC    NL
         JR     Z N_LOOP
         LD     A (HL)
         EX     DE,HL
         POP    HL
         INC    B
         DR     A
         JR     NZ MAIN
         POP    BC
         RET
TEST     CP     123
         RET    NC
         CP     65
         CCF
         RET    NC
         CP     91
         RET    C
         CP     97
         CCF
         RET
```

```
Listing 2
FIND     OR     A
         RET    Z
F_LOOP   BIT    7 (HL)
         INC    HL
         JR     Z F_LOOP
         DEC    A
         JR     NZ F_LOOP
         RET
```

```
Listing 3
JP_TAB   ADD    A A
         LD     C A
         LD     B 0
         ADD    IX BC
         LD     C (IX)
         LD     B (IX+1)
         PUSH   BC
         RET
```

štani u HL i DE registarskom paru. Pre pozivanja rutine TOKEN u HL se smešta adresa početka reči koju treba prepoznati, a u DE adresa tablice. Po povratku iz rutine indikator prenosa je setovan ako je reč prepoznata, a akumulator sadrži redni broj reči u tablici (prva reč ime redni broj 0, druga 1 itd.). Tada je u HL adresa prvog bajta u tekstu posle te reči, tako da je program u mogućnosti da nastavi sa analizom teksta ili eventualnim prikupljanjem argumenata, ako je reč o interpretaciji komandi. Ostali registri se ne prljaju.

Da se ne bi slučajno desilo da se samo deo reči shvati kao neka druga kraća reč, rutina očekuje da prvi simbol iza reči nije slovni, inače nastavlja potragu. Time se sprečava da se, na primer, „DŽINDŽER" protumači kao „DŽIN" u nekom vašem programu koji koristi obe reči.

Sve će biti jasnije posle primera realizacije grananja u željeni potprogram u zavisnosti od reči adresirane sa HL (primer 1). Koristi se i rutina JP_TAB (listing 3) koja skače na adresu na čije mesto u tablici skokova ukazuje akumulator. Pointer na tu tablicu je IX registar. JP_TAB menja još samo BC, tako da se ostali registri mogu iskoristiti za dostavljanje parametara.

Potprogram sa listinga 2 vrši suprotan posao od prve rutine. On nalazi u tabeli adresu reči čiji mu je token dostavljen u akumulatoru. Ovde na tabelu ukazuje HL registar. Dobijena adresa služi da se ta reč prikaže na ekranu ili prenese u neki bafer, tako da se prevođenjem svih šifrovanih reči u tekstu povrati prvobitni sadržaj. Ova rutina se može iskoristiti i za ispisivanje poruka. Recimo, želite da ispišete neki izveštaj o grešci. Tada jednostavno u akumulator smestite kod greške i pozovete rutinu za obradu greške koja sadrži ovaj potprogram i rutinu za ispis niza karaktera. Potpuno analogan potprogram postoji u „spektrumovom" ROM-u.

Na kraju treba reći da su sve ove rutine mašinski nezavisne. Jedini uslov je, naravno, Z80 mikroprocesor.

Branko Marović

```
Primer 1
;HL ukazuje na ime potprograma
;uneto sa tastature
;UNOS, ZAMENA, POTR i
;GREŠKA
;su labele odgovarajućih
;potprograma
         LD     DE T_IME
         CALL   TOKEN
         JR     NC GREŠKA
                        ;nije prepoznato
         LD     IX T_SKOK
         CALL   JP_TAB
T_IME    DEFM   'UNOS
                        ;tablica reči
         DEFB   'S'+128
         DEFM   'ZAMEN'
         DEFB   'A'+128
         DEFM   'NADJ
         DEFB   'I'+128
         DEFB   0 ;kraj
T_SKOK   DEFW   UNOS
         DEFW   ZAMENA
         DEFW   POTR
```

U prošlom broju je u ovoj rubrici prilikom preloma listing isečen i ispremeštan. Tako gornji listing i donji, zaključno sa linijom USER_HL DEFS 2 čini jednu celinu. Sledeće četiri linije pokazuju kako se mogu pozivati rutine iz ROM-a, a rutine FIRM vrši sve to. One se koristi na sledeći način:

CALL FIRM
DEFW adresa

Opis upotrebe ove rutine je ispušten iz teksta što smeta pri čitanju članka.

HELP



Uređuje Žarko Barberski

U svom petom izdanju rubrika „HELP" otvara nove prozore za nove oblike komunikacije među čitaocima. Pitanje *Zdanka Grahovca* koma su potrebne komponente za računar „Galaksija" pokazalo je da su čitaoci „Računara" uvek spremni da svojom solidarnošću pomognu onome ko je u nevolji. Tako se rodila ideja o novom prozoru „Tražimo/nalazimo", koji je namenjen svima onima koji nešto traže (čipove, pločice, razne elektronske komponente, programe, uputstva i sve drugo vezano za računare) i onima koji su našli nešto čime mogu da pomognu onima koji traže. Na našem polu-legalnom piratskom i šverceskom tržištu često je gotovo nemoguće naći ono što vam je neophodno. U takvim uslovima je jedino rešenje da pomognemo jedni drugima u pronalaženju pravih stvari na pravim mestima.

Drugi prozor koji se otvara je „Bajt iz boce", u kome se pojavljuju odgovori na pisma postavljena u „Bajtu u boci". Ovog puta to je pitanje *Saše Tomata*.

Vladimir Kostić je ponudio svoje znanje čitaocima koje zanimaju tajne „Šarp" računara. Njegove odgovore možete očekivati već od sledećeg broja.

Što se slanja pitanja tiče, sve ono što je napisano prethodnih meseci važi i dalje. ŠTO RANIJE POŠALJETE PITANJE, TO SU VEĆE ŠANSE DA NA NJEGA BUDE ODGOVORENO VEĆ U SLEDEĆEM BROJU.

Srećno Programiranje

Bajt iz boce

Odgovor na pitanje *Saše Tomata* (RA 36) u vezi računara „šarp 1350"

Božidar Kostić je poslao prilično iscrpne podatke o FOR/NEXT steku. Pored toga, ponudio je svoju pomoć u odgovaranju na pitanja vezana za „šarp" računare, koju sa zadovoljstvom prihvatamo.

FOR/NEXT stek zauzima 80 bajtova i omogućava realizaciju 5 petlji. Svaka petlja zauzima po 16 bajtova, pri čemu prva dva sadrže apsolutnu adresu (umanjenu za jedan) sadržaja promenljive koja je brojač petlje, sledeće dve sadrže granicu do koje se broji. Vrednost koraka (STEP) nalazi se u prostoru od 8 bajtova neposredno iza granice brojanja. Sledeće 2 bajta sadrže adresu prve naredbe tela petlje i konačno poslednja dva sadrže broj linije od koje počinje telo petlje.

FOR/NEXT stek se može lepo iskoristiti za smeštanje manjih mašinskih programa koji treba da prežive i NEW i resetovanje mašine (ove operacije ne utiču na FOR/NEXT stog). Uslov je, naravno, da bejzik program ne koristi FOR/NEXT petlju.

Drugo pitanje, vezano za predefinisanje seta karaktera, još nije dobilo svoj odgovor. *Jovan Đulajić* je poslao rezultate svog eksperimenta sa sistemskom varijablom 30780 (783C HEX). Setovanjem 6-tog bita ove varijable uključuje se nešto što liči na alternativnu tastaturu sa alternativnim karakter setom. Za sada nije poznato da li se ovo može iskoristiti, jedino se zna da se neke mašine blokiraju posle ove „operacije". Pitanje promene karakter seta ostaje, dakle, i dalje otvoreno.

Božidar Kostić

Bajt u boci

„Atari 130" i 800

Računari „atari 130" i 800 spadaju u retke kućne računare u nas. Već duže vremena u redakciji leže dva neodgovorena pitanja vezana za ove računare.

Bobi Lazareski iz Gostivara ima „atari 130 XE". Nije uspeo da izvrši bejzik program koji sadrži POKE-ove za „Diamonds" (objavljen u RA 30/65) jer dobija grešku u liniji 10. Osim toga, zanima ga kako se radi s POKE-ovima na ovoj mašini, kako da startuje resetovani program, te da li se i kako mogu ovoj mašini dodavati programi na EPROM-ima.

Stjepan Janković iz Ivankova ima „atari 800 XE" i problem sa učitavanjem igara. Računar prvih 3—4 sata odbija da učita igru. Ovakav problem mu se ne javlja kod normalnih programa. Uz čestitke za upornost (3—4 sata učitavanja igra) i zamerku što nije specificirao igre koje mu prave probleme, predajemo ovo pitanje atarijevcima.



Tražimo — nalazimo

Komponente za „galaksiju"

Pitanje *Zdanka Grahovca* kome su potrebna komponente za računar galaksija objavljeno u Računarima 36 doblio je srećan epilog. Javila su se tri čitaoca koji žele da mu pomognu svojim komponentama. Svi oni nude štampanu pločicu tastaturu i druge komponente zaostale iz slavnih dana sklapanja galaksije.
Imena i adrese tih čitalaca su (abecednim redom)

Dic Zoran	Komlošl Leon	Vučković Nikola
Džemala Bijedića 72	Trg 31 divizije 2	Cvijićeva 24
71000 Sarajevo	65282 Cerkno	11000 Beograd
		tel (011) 765-653

Spremnost čitalaca da priteknu u pomoć dala nam je ideju za ovaj novi prozor u kome će se objavljivati pitanja onih koji traže neki čip pločicu program dokumentaciju itd i odgovori svih vas koji želite da svojim informacijama ili materijalom pomognete onome ko je u nevolji

Možda će vam trebati

Kad bejzik preuzme kontrolu

Vlasnici HP-71 mašina koji koriste tekst editor znaju da je ovaj program nemoguće prekinuti ATTN tasterom. Iako je pisan na bejziku. Trik koji je ovde korišćen može se korisno upotrebiti u bilo kom programu koji treba da preuzme celokupnu kontrolu nad mašinom.

Na adresi 2F441 nalazi se fleg koji kada je postavljen (POKE 2F44 'F') onemogućava ATTN taster i to čini sve dok ne bude vraćen na nulu. Ovo međutim ne važi za INPUT Liniju koju on i dalje može poništiti i time posredno izazvati prekid programa. Da bi se i to onemogućilo tasterima ATTN fATTN gATTN USER i 1USER se dodela neki karakteri (na primer DEF KEY f0 C) i to kao izvršne naredbe (sa na kraju). Sada će sa pritiskom na inkriminisane taster INPUT automatski završavati a razlika unetog stringa i DISP$ davaće podatak o tome da li neki od tih tastera pritisnut i koji.

Ovim se trikom mogu različitim tasterima dodeljivati posebna programski kontrolisana „funkcije". Za ilustraciju neka posluže slede-će dve rutine

```
30 POKE 2F441', F' @ ON ERROR GOTO 70
40 READ X
50 DEF KEY '#'&STR$(X) C @ GOTO 40
60 DATA 43 46 99 109 155 165
70 INPUT Isključi me ako možeš X$
80 IF X$= molim te isključi se THEN 90 ELSE 70
90 DISP OK @ POKE 2F441 0 @ PURGE KEYS @ ENG
```

from HP71 Text Editor A

```
250                          LINPUT '  V$ V$ @ P$=DISP$
260  IF  P$#V$  THEN  C$=V$    @ V$=      @ GOTO  290
270 IF NOT LEN(P$) THEN C$=     @ GOTO 320 ELSE M

290 H=POS ( ,UDCTBN  C$[1 1]) @ ON N+1 GOTO
```

Neispravna memorija

Spektrum

Milan Rašić iz Subotice sumnja da mu je memorija u „spektrumu" neispravna. Uradio je prilično opsežan test memorije i zatražio mišljenje naših saradnika

Vaši testovi na žalost nedvosmisleno potvrđuju sumnju da je čitav blok od 32 K RAM-a neispravan pa ga morate zameniti. Memorijske čipove čiju ste identifikaciju naveli možete sasvim slobodno zameniti čipovima 4164.

Voja Gašić

Komodor 64

RND generator

Hasan Bolić koji studira u Zagrebu pravi program za generisanje LOTO kombinacija i treba mu brz i kvalitetan generator slučajnih brojeva

Brzo i kvalitetno generisanje slučajnih brojeva postiže se korišće-njem zvučnog čipa SID (Sound Interface Device). Verovatno će i bejzik rutina

```
k=54286
POKE k 255
POKE k+8 129
POKE k+3 129
```

zadovoljiti vaše potrebe. Po izvršavanju ove rutine sa lokacije 54299 svakim očitavanjem dobijate po jedan slučajan broj u opsegu 0 do 255.

Ukoliko pak želite mašinsku rutinu nastojte da između dva očitavanja bude izvršeno barem nekoliko mašinskih naredbi (ako ništa drugo a ono tri do četiri NOP-a) jer će u protivnom biti generisani mali blokovi istih brojeva budući da SID neće imati dovoljno vremena da između dva očitavanja uvek promeni broj.

O ovoj ideji je pisano još u Računarima 8 ali ju je u međuvreme-nu prekrio veo zaborava

Miroslav Kuzmanović

Knjiga iz izloga

Mira Pavlović iz Zadra traži adresu na koju može poručiti knjigu „Simulacija procesa unutrašnjeg transporta" predstavljenu u „Raču-narima" 38 u rubrici „Izlog knjiga"

Knjigu možete poručiti od izdavača Mašinski Fakultet Univerziteta u Beogradu Izdavački Servis 27 marta 80 11000 Beograd tel (011) 326-651 lokal 316

Zoran Dugarić

YU-slova u GEM-u

Miloš Miladinović iz Beograda ima PC-kompatibilac i radi pod GEM-om, ali ne uspeva da dobije YU-set karaktera

Ukoliko nemate baš ni jedan fajl sa nastavkom FNT možete ga kreirati GEM font-editorom i to slovo po slovo. Možete se naravno obratiti i nekom vlasniku PC računara koji sigurno ima gomilu različitih fontova. Samim kreiranjem fontova posao međutim nije završen. Kreirani font treba još i instalirati. Neki font-editori obavljaju i ovaj posao ali je u principu potrebno imati poseban program za instalira-nje. Problem je zapravo u tome što 1ST WORD nema mogućnost instaliranja vlastitog fonta već mora da koristi onaj iz GEM-a. Pri instaliranju fontova treba paziti i na to da printer zna šta treba da štampa tj potrebni su i odgovarajući drajveri za printer

Amstrad

Disk sa C-128

Igor Nazor iz Splita želi da priključi disk sa C-128 (1571) na „amstrad"

Kalemljenje 1571 diska na amstrad je vrlo skupa i složena operacija pre svega zbog toga što ni C-128 ni amstrad nemaju ugrađena interfejsa za diskove (koji bi im omogućavali da primaju bilo kakvu golu disk jedinicu). Koliko je u beogradskim računarskim krugovima poznato do sada niko još nema rešenja ovog problema.

Glavni kanal za komunikaciju C-182 i amstrada ostaje dakle RS 232. Potrebno je samo nabaviti komunikacione programe za obe mašine što takođe nimalo nije jednostavno

Šarp 1350

Programi i literatura

Saša Tomat iz Ljubljane traži adresu neke firme koja prodaje programe za SHARP-1350

Zapadnonemačka firma Claudius Paschka & Martin Keller Systems, Eifelstrasse 1 6238 Hofheim am Taunus Bundesrepublik Deutschland, Prodaje literaturu i programe za SHARP-1350.

Nenad Vereš

„Atari ST"

Kad padne, zašto je pao i kako ga dići

„Sooner or later it happens to everyone". Ovo nisu stihovi iz pesme Pet Shop Boys-a, nije ni komentar nekog zlurádog i impotentnog čikice, a ni zavidljive usedelice koja posmatra zaljubljene parove. To je, jednostavno, Velika Životna Istina. Ljubav, ljubavni neuspeh, kašalj i ostale velike stvari dešavaju se svima. Programeri doživljavaju još jednu veliku neprijatnost — oni su često žrtve pada sistema.

U gluvo doba noći programer završava program i gledajući ga nežno i očinski smešta ga na disk. Aktivira svoju batch datoteku i žvaćući okoreli sendvič čeka da se završi kompajliranje, optimizacija, linkovanje itd. U glavi simulira rad svojih rutina i zuri u ekran. Gotovo! Skoro pobožno aktivira svoj tek završeni program. Zalogaj mu zastaje u grlu — umesto predivne uvodne slike sleve strane ekrana se pojavljuje niz od sedam bombi a ST odbija da primeti bilo šta što mu sugerišemo sa tastature. Jedino rešenje je reset, povratak u editor i mukotrpno traganje za greškom, dok u glavi odjekuje pitanje „Zašto?" „Zašto?" „Zašto baš meni?"

Kod osmobitnih mašina sve bi bilo izgubljeno, ali Motorola 68000 je dovoljno inteligentna da oseti uslove koji izazivaju pad sistema i da se izvuče ukoliko je moguće. Da bi se ovo postiglo, konstruktori su u čip ugradili osobinu koja se zove „exception processing". U osnovi, ova osobina predstavlja mogućnost da čip prekine bilo šta što radi, uradi nešto drugo i posle toga se vrati na originalni posao. Ovi prekidi se dele na dve grupe: one koji su izazvani spoljašnjim izvorima, kao što su input/output uređaji, i one koji su izazvani unutrašnjim operacijama, kao što su programska greška ili TRAP instrukcije. Spoljašnji prekidi poznatiji su kao „interapti".

Svaki prekid na M68000 obrađuje posebna rutina. Pointeri (vektori) na ove rutine nalaze se u svakom 68000 sistemu u prvih 1024 bajta memorije. Vektori se tu nalaze u obliku duge reči (četiri bajta) što znači da ih ima 256, a obeleženi su brojevima od 0 do 255.

U sledećoj tabeli dati su brojevi i adrese svih vektora.

VEKTOR	ADRESA	FUNKCIJA
0	$000	Stack pointer after reset
1	$004	Program counter after reset
2	$004	Bus error
3	$00C	Address error
4	$010	Illegal instruction
5	$014	Division by zero
6	$018	CHK instruction
7	$01C	TRAPV instruction
8	$020	Privilege violation
9	$024	Trace
10	$028	Line A emulator
11	$02C	Line F emulator
12—14	$030—$038	reserved
15	$03C	Uninitialized interrupt
16—23	$040—$05C	reserved
24	$060	Spurious interrupt
25—31	$064—$07C	Level 1—7 interrupt
32—47	$080—$0BC	TRAP 0—15 Instruction
48—63	$0C0—$0FC	reserved
64—255	$100—$3FC	User interrupt vector



Od ovih vektora ST koristi sledeće

VEKTOR	FUNKCIJA
10	Line A emulator
26	Level 2 interrupt
28	Level 4 interrupt
33	TRAP 1 GEMDOS
34	TRAP 2 GEM
45	TRAP 13 BIOS
46	TRAP 14 XBIOS

Vektore 64—79 koristi MFP 68901 za svoje interapte. Svi neupotrebljeni vektori su na raspolaganju korisniku, na primer 12 neupotrebljenih TRAP-ova.

Ako ST počne da pada, M68000 će primetiti da nešto nije u redu i odmah upisati vrednosti data i adress registara u nisko područje u memoriji koje obično preživljava krah, a računar posle reseta tu ništa ne piše. Neku indikaciju o prirodi pada sistema daje i sam računar, jer broj bombi odgovara broju vektora koji je prekid izazvao.

Reset: Vektori 0 i 1 se koriste pri uključivanju i resetu računara za inicijalizaciju stak pointera i programskog brojača.

Bus error Vektor 2 pokazuje da je program pokušao da pristupi nepostojećoj memoriji.

Address error Vektor 3 može biti izazvan pokušajem pristupa dugoj reči u memoriji na neparnoj adresi.

Illegal instruction Vektor 4 pokazuje da vaš program pokušava da izvrši instrukciju koja se ne nalazi u skupu instrukcija procesora M68000 i najčešće je izazvan greškom u kucanju (u asembleru).

Division by zero Vektor 5 je izazvan deljenjem nulom što je matematički pogrešno.

CHK instruction Vektor 6 je izazvan CHK instrukcijom.

TRAPV instruction Vektor 7 je izazvan TRAPV instrukcijom.

Privilege violation Vektor 8 je prouzrokovan upotrebom privilegovane instrukcije dok M68000 nije bio u supervisor modu.

Trace Vektor 9 se upotrebljava uglavnom u dibagerima gde je potrebno debagirati program korak po korak.

Line A emulator Vektor 10 je trap za op-kodove u formatu „$Axxx".

Line F emulator Vektor 11 je trap za op-kodove u formatu „$Fxxx".

Pored očitavanja vektora prekida moguće je iz ST-a pročitati i druge podatke koje je računar upisao u vreme kraha.

Adresa $0380 sadrži broj $12345678. Ukoliko se na tom mestu nalazi drugi broj, informacija o krahu nije sačuvana.

Data registri D0-D7 su sačuvani počevši od lokacije $0384.

Adresni registri A0—A6 i supervizor stek pointer A7 su sačuvani počevši od adrese $03A4.

Vektor prekida je na adresi $03C4 kao duga reč, a korisnički registar na $03C8.

Počevši od $03CC nalazi se lista od 16 reči na koje je pokazivao supervizor stek.

Naravno, sve ovo važi samo u slučaju da ST nije u međuvremenu isključivan.

Ove lokacije je vrlo teško ispisati bez nekog monitora ili dibagera, ali rešenje postoji. Melodramu s početka teksta možemo vrlo lako razrešiti programom „Crash Analyzer" objavljenim još u oktobarskom broju časopisa „COMPUTE!'s Atari ST Disk & Magazine" 1986-te godine. Odman posle reseta treba aktivirati program ANALYZER.TOS i na ekranu će se pojaviti registri D0—D7, A0—A7, vektor prekida, lista flegova setovanih u trenutku kraha, startne adrese supervizor i korisničkog steka, kao i sadržaj supervizor steka u trenutku kraha.

Informacije dobijene od ovog programa uz poznavanje programiranja M68000 pomoći će vam da lakše ispravite grešku u programu ili bar steknete utisak o tome gde ste pogrešili.

Podaci potrebni za pisanje ovog teksta, kao i tabela vektora prekida, uzeti su iz knjige „ATARI ST ANATOMY" u izdanju kuće „First Publishing Ltd".

Dušan Mikulić

Neka bude borba neprestana

Priča o događaju koji je inspirisao nastanak „korvora" je uglavnom već poznata: jedan programer, u jednom centru, napisao je jedan program koji se duplicirao svakim svojim izvršavanjem. Taj program, inače nazvan „Creeper", polako je „grickao" memoriju, pa je ceo sistem bio pred raspadom. Epilog se dao naslutiti, ali . . . Neki pametni programer se dosetio i napravio jedan drugi program nazvan „Reaper", čiji je zadatak bio takođe da se multiplicira, ali i da pri tome uništava kopije „Creeper"-a koje su se nalazile u memoriji. Pošto je izvršio zadatak, „Reaper" je uništio i samog sebe, i sistem se vratio u normalan režim rada. Četvrti nastavak naše serije o „korvoru" posvećujemo programima — borcima.

U broju 36 je objavljen listing MARS-a koji prema svim Marfijevim zakonima nije mogao biti sasvim korektan. Iako greške nisu bile suštinske prirode, one u najvećem broju slučajeva onemogućavaju pravilan rad simulatora.

Prva greška se nalazi u proceduri ***input files*** i to u instrukciji ***readln (file1)***. Umesto toga treba da stoji ***readln (file2)***.

Druga greška se ponavlja u procedurama ***mem-scan*** i ***input-files***, gde umesto izračunavanja po modu 2000 treba da stoji po modu 8000. Osim toga, u proceduri ***input-files***, umesto instrukcije ***p =random(2000)*** treba da bude ***p =random(8000)***.

I što se tiče grešaka, to bi bilo sve.

Da bi slika „korvora" bila potpuna, za ovaj broj je pripremljen i editor za unos programa u fajlove odakle ih kasnije MARS poziva. Po nekoj analogiji, ovaj program je nazvan ARES.

Iako se i iz samog listinga može zaključiti princip rada programa, ovde ćemo ipak ukazati na neke njegove najosnovnije karakteristike.

Program sadrži 12 opcija za rad, od kojih jedna, bolje rečeno, služi za prekid rada ***(quit)***. Program-borac se prethodno unosi u bafer, odakle se opcijom ***store file*** prenosi u fajl. Već prenet program moguće je ponovo vratiti u bafer opcijom ***take file***, gde može biti naknadno editovan.

Unos programske linije se vrši opcijom ***input line***, a njeno brisanje opcijom ***erase line***.

Kretanja pointera po baferu je omogućeno sa ***back*** i ***forward***. Pointer se postavlja na početak bafera sa ***reset pointer***, a na kraj bafera sa list. Pri tome se listing sprovodi od programske linije na kojoj se nalazio pointer, pa do kraja bafera. Inače, bafer je dugačak onoliko koliko i sam uneti program (maksimalno 1000 linija). Bafer se prazni sa ***clear buffer***, a broj programskih linija koje je još moguće uneti se učitava opcijom ***number of free locations***.

Jedna važna napomena prilikom rada sa editorom.

Unos programske linije se vrši iza linije na koju pokazuje pointer (poslednja prikazana linija na ekranu). Pri tome nije obavezno da se unos vrši iza poslednje unete linije, već može i u sredini programa, pri čemu se ostale pomeraju za jedno mesto unapred. To znači da će, ukoliko je program veće dužine od 1000 linija, a unos se vrši negde u sredini, zadnja linija biti obrisana, tj. „ispašće" iz bafera.

Osim toga, pomoću erase line briše se linija na koju pokazuje pointer, ali pri tome ostale instrukcije popunjavaju njeno mesto, ukoliko je ona bila negde unutar programa, tako da u programu ne ostaje „prazan prostor".

Time bi prikaz editora bio završen. Iako se editor može nalaziti kao procedura u okviru MARS-a, pogodnije je da stoji kao zaseban program, jer su njihove funkcije u okviru „korvora" ipak jasno odeljene.

```
„Dwarf":            „Imp":
ADD #5,3            MOV 0,1
MOV #0,Q2
JMP —2              „Vantage":
DAT — 4             ADD # 15 10
                    MOV # 0 Q9
„Gemini":           MOV Q11 Q10
MOV Q10 Q9          CMP 10,8
CMP 9,7             JMP 4
JMP 4               SUB # 1 7
SUB # 1 6           SUB # 1 7
SUB # 1 6           JMP — 7
JMP — 5             MOV # 211 214
MOV # 101 103       JMP 201
JMP 93              DAT — 11
DAT — 10            DAT — 13
DAT 101             DAT 211
DAT 0               DAT 0
```

Sl. 1. Programi-borci i njihovi kodovi

Mali glupi . . .

Vratimo se sada samom „korvoru". Na sl. 1 su data četiri programa „Dwarf", „Imp", „Gemini" i „Vantage". Iako ste se s nekima već ranije upoznali, izvršićemo njihovu međusobnu komparaciju.

„Dwarf" je izuzetno glup, ali u isto vreme i izuzetno opasan borac. On „bombarduje" svaku petu memorijsku lokaciju, a za to mu je potrebno svega 3 ciklusa (ciklus odgovara jednoj izvršenoj instrukciji). Program koji želi da mu se suprodstavi ima svega tri mogućnosti: da ga prvi pogodi, da se kreće po memoriji i tako izbegava „bombe", ili da apsorbuje pogotke i popravlja oštećene segmente.

Prva strategija je zasnovana na slučajnom pogotku, jer je „Dwarf" dugačak svega 4 linije (u stvari 3, jer mu DAT instrukcija nije od prevelikog značaja), pa je potrebno oko 2000 „gađanja" da bi se on uništio. U takvom dvoboju bi uglavnom „Dwarf" izlazio kao pobednik, jer je nemoguće napraviti kraći i efikasniji stacionarni program nego što je on. Zanimljiv je sukob dva „Dwarf"-a. U njemu svaki „Dwarf" ima 30% šanse da pobedi, dok je 40% verovatnoća da nijedan od njih neće zadati fatalan udarac drugom u nekom određenom vremenskom periodu.

Pre nego što razmotrimo druge dve strategije, podsetimo se „Imp"-a, koji je najprostiji program pisan u „redkodu". Šta se dešava ako „Imp" suprodstavimo „Dwarf"-u? Iako se „baražna vatra" „Dwarf"-a kreće brže nego „Imp", to ipak ne daje neku veliku prednost, jer je šansa da se tada „Imp" pogodi svega 20%. Ukoliko međutim „Imp" sustigne „Dwarf"-a pre nego što ga ovaj pogodi, on će se prosto „ušetati" u njegovu strukturu, tako da ćemo kao rezultat dobiti novi „Imp". A prema pravilima „korvora", tada je rezultat borbe nerešen.

Oba programa, i „Dwarf" i „Imp", predstavljaju klasu programa za koje se može reći da su mali, agresivni, ali i glupi.

. . . oni između . . .

Višu klasu predstavljaju programi koji su nešto duži, malo manje agresivni, ali dovoljno pametni da izađu na kraj sa programima iz prethodne kategorije. Svaki od tih programa sadrži segment koda sličan onom u programu „Gemini".

Program „Gemini" se sastoji iz tri dela: iz petlje u kojoj se izvršava kopiranje sopstvenih instrukcija, iz DAT instrukcija koje indiciraju sledeću instrukciju koja će se kopirati i njeno odredište, i iz finalne JMP instrukcije koja predaje kontrolu procesa kopiji. Na taj način će program dugačak 11 linija, nakon 54 ciklusa biti prekopiran 100 memorijskih lokacija dalje.

Modifikovanjem programa „Gemini" moguće je dobiti programe iz iste klase. Jedan od njih je i „Bigfoot", koji se od „Gemini"-ja razlikuje samo u dužini skoka. Naime, kod njega je interval između kopija jednak nekom velikom prostom broju, tako da ga je izuzetno teško pogoditi periodičnim gađanjem (programi su uglavnom takve koncepcije da periodično šalju DAT „bombe".)

U suštini i „Gemini" i „Bigfoot" predstavljaju međuklasu programa, jer njihova koncepcija nije dovoljno razvijena za prave borbene programe. Njihova agresivnost se svodi samo na to da svojim kopiranim segmentima poremete koncepciju suparnika i da ga tako navedu na neku već postavljenu DAT instrukciju.

Pravi predstavnik druge klase programa bi bio na primer „Vantage", koji pored toga što se kopira 210 instrukcija unapred, u svaku petnaestu memorijsku lokaciju ubacuje po jednu DAT instrukciju. Njegove perfomanse su:

- dužina programa: 14 programskih linija
- dužina skoka: 210 memorijskih lokacija
- brzina prenosa: 98 ciklusa
- brzina gađanja: 13/98 „bombi" po ciklusu
- periodičnost gađanja: 15 memorijskih lokacija

Specifičnost „Vantage"-a je u tome što se morala uskladiti dužina skoka i brzina prenosa sa brzinom i periodičnošću gađanja, kako program ne bi sam sebe uništio. Pri drugim vrednostima parametara ubrzo bi došlo do samouništenja programa.

Iako je „Vantage" ubedljivo najbolji od svih do sada prikazanih programa, ni za njega se ne može reći da je naročito pametan. Naime, i on, kao i prethodni, ima unapred determinisano ponašanje — ako se u jednom trenutku zna faza izvršenja programa, moguće ju je odrediti i u bilo kom drugom.

. . . i veliki pametni programi

Međutim, postoji i kategorija programa kojima se ne može tako jednostavno predvideti

```
END;

PROCEDURE back;
BEGIN
 IF p<>0 THEN BEGIN
            p := p-1;
            IF p<>-1 THEN print (p)
                    ELSE writeln('Top of buffer.')
          END
END;

PROCEDURE for_ward;
BEGIN
 IF p<sub THEN BEGIN
             p := p+1;
             print (p)
            END
        ELSE writeln ('Bottom of buffer.')
END;

PROCEDURE erase_line;
VAR  i : integer;
BEGIN
 IF p<>0 THEN BEGIN
            sub := sub-1;
            FOR i:=p TO sub DO buf[i] := buf[i+1];
            IF p>sub THEN p := sub;
            IF p<>0 THEN print (p)
                    ELSE writeln ('Top of buffer.')
          END
END;

PROCEDURE input_line;
VAR
     i : integer;
     mn : str;
     enic : mem_loc;
     error : boolean;
BEGIN
 IF p=999 THEN writeln ('Bottom of buffer.')
       ELSE BEGIN
             WITH enic DO BEGIN
                         code := '0';
                         arg_A.adr_type := ' ';
                         arg_A.value := 0;
                         arg_B.adr_type := ' ';
                         arg_B.value := 0
                        END;
             error := FALSE;
             read (mn);
             IF mn='DAT' THEN edit.code := '0' ELSE
             IF mn='MOV' THEN edit.code := '1' ELSE
             IF mn='ADD' THEN edit.code := '2' ELSE
             IF mn='SUB' THEN edit.code := '3' ELSE
             IF mn='JMP' THEN edit.code := '4' ELSE
             IF mn='JMZ' THEN edit.code := '5' ELSE
             IF mn='JMG' THEN edit.code := '6' ELSE
             IF mn='DJZ' THEN edit.code := '7' ELSE
             IF mn='CMP' THEN edit.code := '8' ELSE
             error := TRUE;
             read (edit.arg_A.adr_type);
             IF (edit.arg_A.adr_type<>'#') AND (edit.arg_A.adr_type<>'@')
                AND (edit.arg_A.adr_type<>' ') THEN error := TRUE;
             read (edit.arg_A.value);
             edit.arg_A.value := edit.arg_A.value MOD 8000;
             read (edit.arg_B.adr_type);
             IF (edit.arg_B.adr_type<>'#') AND (edit.arg_B.adr_type<>'@')
                AND (edit.arg_B.adr_type<>' ') THEN error := TRUE;
             read (edit.arg_B.value);
             edit.arg_B.value := edit.arg_B.value MOD 8000;
             readln;
             IF error=TRUE THEN writeln ('Syntax error.')
                          ELSE BEGIN
                                 p := p+1;
                                 IF sub<999 THEN sub := sub+1;
                                 FOR i:=sub DOWNTO p+1 DO buf[i]:=buf[i-1];
                                 buf[p] := enic
                               END
            END
END;

PROGRAM ARES;

TYPE
     argument = RECORD
                  adr_type : char;
                  value    : integer
                END;
     mem_loc  = RECORD
                  code          : char;
                  arg_A, arg_B  : argument
                END;
     str      = STRING [3];

VAR
     buf : ARRAY [0..999] OF mem_loc;
     p, sub : integer;
     instruction : char;

FUNCTION mnemonic (i:integer) : str;
BEGIN
 CASE buf[i].code OF
     '0' : mnemonic := 'DAT';
     '1' : mnemonic := 'MOV';
     '2' : mnemonic := 'ADD';
     '3' : mnemonic := 'SUB';
     '4' : mnemonic := 'JMP';
     '5' : mnemonic := 'JMZ';
     '6' : mnemonic := 'JMG';
     '7' : mnemonic := 'DJZ';
     '8' : mnemonic := 'CMP'
 END
END;

PROCEDURE print (i:integer);
BEGIN
 writeln (buf[i].code,    mnemonic(i),
          buf[i].arg_A.adr_type, buf[i].arg_A.value,
          buf[i].arg_B.adr_type, buf[i].arg_B.value)
END;

PROCEDURE take_file;
VAR
     f : FILE OF mem_loc;
     fname : STRING [12];
BEGIN
 writeln;
 write ('File of program : ');
 readln (fname);
 assign (f,fname);
 reset (f);
 p := 0;
 WHILE NOT eof(f) DO
  BEGIN
   read (f,buf[p]);
   print (p);
   p := p+1
  END;
 sub := p;
 writeln;
 writeln ('Loading is over.');
 writeln
END;

PROCEDURE rewind;
BEGIN
 p := -1;
 writeln;
 writeln ('Buffer-pointer is reset.');
 writeln
END;

PROCEDURE list;
VAR  i : integer;
BEGIN
 writeln;
 FOR i:=p+1 TO sub DO print (i);
 p := sub;
 writeln;
 writeln ('Listing is over.');
 writeln

PROCEDURE store_file;
VAR
     i : integer;
     f : FILE OF mem_loc;
     fname : STRING [12];
BEGIN
 writeln;
 write ('File of program : ');
 readln (fname);
 assign (f,fname);
 reset (f);
 FOR i:=0 TO sub DO write (f,buf[i]);
 writeln;
 writeln ('Program is stored in file.');
 writeln
END;

PROCEDURE clear_buffer;
BEGIN
 p := -1;
 sub := -1
END;

PROCEDURE quit;
BEGIN
 writeln;
 writeln;
 writeln ('Good bye.')
END;

PROCEDURE number;
BEGIN
 writeln;
 writeln ('Number of free locations : ',999-sub);
 writeln
END;

PROCEDURE helps;
BEGIN
 writeln;
 writeln;
 writeln ('Options :');
 writeln;
 writeln ('H - help');
 writeln ('B - back');
 writeln ('F - forward');
 writeln ('I - input line');
 writeln ('E - erase line');
 writeln ('L - list');
 writeln ('R - reset pointer');
 writeln ('C - clear buffer');
 writeln ('N - number of free locations');
 writeln ('S - store file');
 writeln ('T - take file');
 writeln ('Q - quit');
 writeln
END;

BEGIN
 clear_buffer;
 helps;
 REPEAT
  read (kbd,instruction);
  CASE instruction OF
      'H' : helps;
      'B' : back;
      'F' : for_ward;
      'I' : input_line;
      'E' : erase_line;
      'L' : list;
      'C' : clear_buffer;
      'N' : number;
      'S' : store_file;
      'T' : take_file;
      'Q' : quit
  END
 UNTIL instruction='Q'
END.
```

ponašanje, jer ono u najvećoj meri zavisi od ponašanja njihovih suparnika. Kako su oni, po prirodi, dosta dugački, ovde će biti samo uopšteno prikazani.

Jedan od njih se zove „Raidar" i sadrži dva „piketa" koji ga okružuju sa obe strane (sl.2.). Svaki piket se sastoji iz 100 uzastopnih memorijskih lokacija napunjenih jedinicama koje su odvojene od glavnog dela programa bafer-zonom od 100 praznih memorijskih lokacija. Na principu prostog tajm-šeringa „Raidar" periodično bombarduje udaljena memorijske zone i proverava stanje piketa. Ukoliko je sadržaj lokacija piketa promenjen, „Raidar" to registruje kao napad „Dwarf"-a, „Imp"-a ili nekog drugog programa. Nakon toga, „Raidar" sam sebe kopira na drugu stranu oštećenog piketa, popravlja oštećenja na njemu, konstruiše novi piket na nezaštićenoj strani i nastavlja sa uobičajenim protokolom. Kako postoji mala verovatnoća da bude pogođen glavni deo programa, a da napad ne bude ranije signaliziran preko piketa, može se reći da je ovakva koncepcija programa dosta dobra. Međutim, šansa za njegov poraz ipak postoje.

David Jones (University of Western Ontario) je dizajnirao program nazvan „Scanner", koji je bio u stanju da na samome sebi uklanja nedostatke prouzrokovane napadima suparnika. „Scanner" sadrži dve identične kopije, od kojih se interpretira uglavnom samo jedna. Ta kopija periodično skanira sadržaj druge, s namerom da otkrije da li neka instrukcija nije možda promenjena. Ukoliko jeste, ona se zamenjuje korektnom i kontrola se predaje drugoj kopiji. Takav program uspešno apsorbuje pogotke „Dwarf"-a, „Imp"-a i sličnih sporo-napredujućih suparnika. Međutim, za ozbiljniju zaštitu takvog sistema, potrebno je tri, pa i više integrisanih kopija (usput rečeno, većina „vojadžera" je imala takvu zaštitu programa, ali ne usled opasnosti od „virusa", nego od kosmičkog zračenja).

Nedostatak ovih programa je glomaznost, pa se strategija apsorbovanja pogodaka u ovom slučaju pokazala i suviše defanzivnom. Razne ideje (od kojih je većina još uvek u eksperimentalnoj fazi) pokušava da reši problem glomaznosti sofisticiranih programa. Jedna od njih je bila da se proširi set instrukcija tako da se olakša mogućnost zaštite. Tako, na primer, instrukcija PCT A bi štitila sadržaj memorijske lokacije određena argumentom A za vreme izvršavanja naredna instrukcija. No, time problem u svakom slučaju ne bi bio do kraja rešen.

Korvor polako dostiže svoju kulminaciju.

Umesto kraja

Priča o „Creeper"-u i „Reaper"-u je zasnovana na elementima dva već odavno postojeća programa. Jedan program je kompjuterska igra zvana „Darwin", koju je dizajnirao Douglas M. McIlroy (AT&T Bell Laboratories), a drugi je eksperimentalni program „Worm", čiji je autor John F. Shoch (Xerox Palo Alto Research Center).

U igri „Darwin" svaki igrač poseduje određeni broj „organizama" programa napisanih na asembleru koji se nalaze u memoriji zajedno sa „organizmima" drugih igrača. Cilj igre je da „organizmi" koje je kreirao jedan igrač (dakle, koji pripadaju jednoj „vrsti") unište „organizme druge vrste" i zauzmu njihovu teritoriju. „Pobednik je onaj igrač čiji su „organizmi" najbrojniji nakon određenog vremenskog perioda. Zanimljivo je da je McIlroy pronašao „neuništiv organizam" koji iako „besmrtan", nije pokazivao previše agresivnosti.

Program „Worm" je dizajniran da omogući maksimalnu moguću iskorišćenost minikompjutera povezanih u Xerox mrežu. Preko programa-supervizora „Worm" se unosi u svaki kompjuter u mreži tako da koordinira rad svih kompjutera prilikom rešavanja određenog kompleksnog problema. Pri tome održava ravnomernu raspodelu radnih zadataka u mreži i omogućava istupanje pojedinog kompjutera iz već postojećeg radnog režima, ukoliko korisnik tako želi.

Iako je korelacija očigledna, dalje od imaginacije pomenutih elemenata se u tim programima nije otišlo. Ali u „Korvoru" „Creeper" i „Reaper" su postali stvarnost.

Dragan Stanišić

Bulova algebra/Turbo Pascal 4.0

Analizator Bulovih iskaza

U programiranju je često potrebno ispitivati kombinacije složenih uslova. Program „Analizator Bulovih iskaza" za uneti iskaz crta tabelu sa svim kombinacijama vrednosti za promenljive i odgovarajućim vrednostima iskaza. Osim obrazovnog karaktera, ovaj program može biti od velike pomoći pri ispitivanju iskaza drugim metodama (za iskaze sa 5 ili više promenljivih tabela ima 2^5=32 ili više redova, što je suviše veliki posao za čoveka), kada na osnovu računareve tabele možemo da proverimo tačnost rezultata.

Po definiciji Bulova algebra je struktura koju čini skup B sa svoja dva elementa (0,1), dve binarne operacije (presek, unija) i jednom unarnom (komplement) ako su ispunjene aksiome Bulove algebre (koje nema potrebe da navodim). Operativno se umesto operacija *unija, presek* koriste *i, ili*, a umesto 1 i 0 *true i false*. Pošto kompjuter ne poseduje u osnovnom setu karaktere znaka za operacije *i, ili* i *komplement (negacija)* u programu se koriste simboli • + i N (strogo veliko N a ne malo). Osim ovih operacija mogu se koristiti i zagrade i promenljive (p, q, r ... isključivo mala slova).

Jedini problem u ovakvim programima predstavlja pisanje funkcije za izračunavanje iskaza za data vrednosti promenljivih.

Vrednosti iskaza

Zamislite da je potrebno izračunati vrednost iskaza u kojem se pojavljuju samo prirodni brojevi i aritmetičke operacije sabiranja i oduzimanja. Prvo definišemo dva niza **br** i **op** (dimenzije nisu važne). U petlji iz iskaza naizmenično „vadimo" broj (funkcija VAL) i operaciju (+ ili −) i stavljamo ih u odgovarajuće nizove **br** i **op**. Zatim u drugoj petlji ispitujemo o kojoj se operaciji radi (recimo, ispitujemo **op[1]**) i rezultat (**br[1]+−br[2]**) smeštamo u varijablu niza **br** u kojoj je bio prvi sabirak (u **br[1]**). Pomeramo elemente niza (šiftujemo nizove) **br** za jedno mesto ulevo počevši od elementa iza drugog sabirka (**br[2] =br[3], br[3] =br[4]** ...) i niza **op**, počevši od elementa iza operacije koje je upravo izvršena (**op[1] =op[2], op[2] =op[3],** ...).

Međutim, sada uvodimo i operacije množenje i deljenja (koje imaju veći prioritet od + −). Prva petlja ostaje ista. Međutim u drugoj petlji prvo izvršavamo operacije • i /, pa tek onda operacije + i −. Uvođenje zagrada je na prvi pogled veliki problem ali ... izraz u zagradi u stvari zamenjuje jedan broj rezultat tog iskaza tako da u (prvoj) petlji u kojoj „vadimo" brojeva i operacije moramo prvo da izračunamo vrednost iskaza u zagradi pa tek onda smestimo konkretan broj (rezultat iskaza u zagradi) u niz **br**. Važno je primetiti da je proces izračunavanja iskaza u zagradi isti kao i proces izračunavanja „glavnog" iskaza (oba iskaza su sastavljena od istih elemenata: operacije i brojeva ili zagrada). Prema tome, iskaz u zagradi ćemo izračunati tako što će funkcija rekurzivno pozvati samu sebe, ali ovaj put će argument biti iskaz u zagradi (funkcija vraća rezultat iskaza).

Da se sad vratimo na Bulove iskaze koji imaju sličnu konstrukciju. Glavna razlika je što umesto konkretnih brojeva sada imamo nizove vrednosti koje promenljive uzimaju. Zato se rezultat ne može vratiti kao rezultat funkcije, već se prenosi kao parametar procedure **po vrednosti**. Evo i liste prioriteta operacija:

1. **N** (negacija)
2. **I** (konjunkcija), **ili** (disjunkcija)
3. => (implikacija), <=> (ekvivalencija)

Implikacija i ekvivalencija su izvedene operacije i mogu se zameniti sledećim formulama: p=>q <=> N(p•Nq) <=> (de Morganova pravila) Np+q, p<=>q <=> (p•q)+(Np•Nq).

Funkcija *sintaksa*

Program počinje unosom znakova koji treba da predstavljaju **true** i **false**. Pri tom se koristi nestandardna funkcija **ReadKey** (isto kao i INKEY$ u spektrumovom bejziku), ali se ona međutim slobodno može zameniti **Readln** procedurom. Zatim se unosi iskaz sve dok funkcija **sintaksa** ne vrati vrednost **true**.

Dakle, funkcija **sintaksa** kao argument ima iskaz a vraća **false** ako ustanovi da iskaz nije ispravan. Varijabla **ras** sadrži rezultat funkcije i na početku je **true** (pretpostavljamo da je iskaz ispravan). Brojač n sadrži redni broj karaktera u iskazu koji se ispituje. Treba imati na umu da „brojevi" (argumenti operacija) mogu biti promenljiva (1 znak), ali i kombinacija negacija i izraza u zagradama (više znakova) pri čemu moramo voditi računa da broj otvorenih zagrada mora biti jednak broju zatvorenih zagrada. Pošto je broj operacija uvek za jedan manji od „brojeva", ispred petlje ispitujemo prvi „broj", pa tek onda u petlji ispitujemo naizmenično operaciju pa broj sve dok ne ispitamo ceo izraz ili naiđemo na grešku.

Pri ispitivanju karaktera iskaza koriste se skupovi. Tako uslovni izraz **s[n] in ['(', 'N']** radi isto što i **(s[n]= '(') or (s[n]= 'N')** — ispituje da li je karakter **s[n]** jednak nekom od karaktera koji su navedeni pod navodnicima između uglastih zagrada, u ovom slučaju (,N. Takođe, kod ispitivanja da li je deo stringa <=>/=> (operacije ekvivalencije i implikacije) nema potrebe da se ispituju sve tri/četiri znaka, već je dovoljno ispitati samo prvi znak. Na IBM PC-u se aski kodovi malih slova nalaze u intervalu 97—122, pa pri prekucavanju listinga proverite da li isto važi i za vaš računar.

Kada funkcija **sintaksa** vrati vrednost **true**, računar izlazi iz petlje. Prvo se u iskazu pronalaze varijable tako što se za svaki znak ispituje da li je malo slovo. Ako jeste, ispituje se da li ta promenljiva već postoji u nizu **s** i ako ne postoji pamti se karakter (ime promenljive) i povećava odgovarajući brojač. Zatim se u petlji na osnovu broja promenljivih računa broj vrednosti koje svaka promenljiva uzima (br_vre=2^br_name). Potom procedura **poc_vrednosti** definiše vrednosti promenljivih (sve moguće kombinacije). Pošto se niz **a** ne pojavljuje kao argument u pozivu procedure, a koristi se u samoj proceduri, sve promene vrednosti elemenata niza **a** u proceduri bivaju promenjene i u glavnom programu.

Procedura *calculate*

Procedura **calculate** (koja obavlja računanje vrednosti izaza) poziva se sa sledećim parametrima: **izraz** je string koji sadrži iskaz koji treba da se izračuna i **raz** je niz koji se prenosi **po vrednosti,** jer će u njemu biti vraćeni rezultati iskaza u glavni program (ili proceduru rekurzije). Struktura podataka procedura je sledeća.

Procedura je podeljena u dva dela. U prvom delu (**while** petlja) se iz iskaza „vade brojevi" (promenljive iz glavnog programa ili rezultati iskaza u zagradi) i operacije u nizove **br** i **op**. Varijabla **n** označava redni broj karaktera u stringu **izraz** koji se obrađuje. Vađenje brojeva sadrži sledeće operacije:

— ako je ispred „broja" N, tada varijabla negacije postaje **true**

— ako broj počinje otvorenom zagradom, traži se **odgovarajuća** zatvorena zagrada i rekurzivno se poziva procedura **calculate**, ali ovog puta su argumenti iskaz u zagradi i podniz niza **br** gde se vraća rezultat iskaza u zagradi (nulti

Konstante	
maxlen	maksimalna dužina iskaza (karaktera)
maxels	maksimalan broj promenljivih
maxvre	maksimalan broj vrednosti promenljivih 2^10=1024
Tipovi	
izrazt	sa sam iskaz koji se računa
basic	tip promenljivih sadrži 2 polje
	name je oznaka sama promenljive (p, q, r, ...)
	value vrednosti promenljive
Promenljive	
n, m, t	su najčešće brojači
br_name, br_vre	broj varijabli i broj vrednosti koja uzima svaka varijabla (br_vre=2^br_name)
tačne, netačne	znaci za vrednosti true i false 1 i 0 u tabeli
a	sadrži sve promenljive
izraz	(glavni) iskaz
tautologija	ako je true onda je iskaz tautologija (znači da je za bilo koje vrednosti promenljivih iskaz uvek tačan)
rez	niz sa rezultatima
br	niz sadrži vrednosti promenljivih (iz glavnog programa istog je tipa brt) ili vrednosti iskaza u zagradi (rekurzije)
op	niz sadrži operacije (u slučaju <=> i => uzima se prvi znak
n, m, t	brojači
br_zagrada	broj zagrada
br_br	brojač „brojeva" (nema potrebe da se uvodi i **br_op** (brojač operacija) jer je uvek **br_op=br_br−1**

```
program analizator_bulovih_izraza(input,output);
uses crt;

const maxlen=150;
      maxvre=4096;
      maxslo=10;

type izrazt=string[maxlen];
     brt=array[1..maxvre] of boolean;
     basic=record name:char;
                  value:brt;
     end;

var m,n,t,br_name,br_vre:integer;
    odgovor,tacno,netacno:char;
    s:array[1..maxslo] of basic;
    izraz:izrazt;
    nepostoji,tautologija:boolean;
    rez:brt;

function citaj:char;
var rez:char;
begin
  repeat
    rez:=UpCase(ReadKey)
  until rez in ['D','N'];
  writeln(' ',rez);
  citaj:=rez;
end;

function sintaksa(s:izrazt):boolean;
var rez:boolean;
    n,br_zagrada:integer;
begin
  rez:=true;n:=1;br_zagrada:=0;
  while s[n] in ['(','#'] do
  begin
    if s[n]='(' then br_zagrada:=br_zagrada+1;
    n:=n+1;
  end;
  if (ord(s[n])<97)or(ord(s[n])>122) then rez:=false;
  n:=n+1;
  repeat
    if not(s[n] in ['+','*','=','<']) then rez:=false
    else
    begin
      case s[n] of
        '+','*':n:=n+1;
        '=':n:=n+2;
        '<':n:=n+3
      end
    end;
    while s[n] in ['(','#'] do
    begin
      if s[n]='(' then br_zagrada:=br_zagrada+1;
      n:=n+1
    end;
    if (ord(s[n])<97)or(ord(s[n])>122) then rez:=false;
    n:=n+1;
    while s[n]=')' do
    begin
      n:=n+1;br_zagrada:=br_zagrada-1;
    end
  until not(rez)or(n>length(s));
  if br_zagrada<>0 then rez:=false;
  sintaksa:=rez
end;

procedure poc_vrednosti(br:integer);
var m,n,h,t:integer;
begin                        {Postavlja pocetne vrednosti osnovnih slova?}
  t:=1;
  for n:=br_name downto 1 do
  begin
    for h:=1 to trunc(br/(t*2)) do
    begin
      for m:=1 to t do
        s[n].value[(h-1)*t*2+m]:=false;
      for m:=1 to t do
        s[n].value[(h-1)*t*2+t+m]:=true
    end;
    t:=t*2
  end;
end;

procedure calculate(izraz:izrazt;var rez:brt);
const max_br=15;
var  br:array[1..max_br] of brt;
     op:array[1..max_br] of char;
     n,m,t,br_zagrada,br_br:integer;
     negacija:boolean;
     izraz1:izrazt;
begin
  n:=1;negacija:=false;br_br:=1;
                                {Vadjenje brojeva i operatora iz izraza!}
  while n<=length(izraz) do
  begin
    if izraz[n]='#' then        {Prvo brojevi}
    begin
      negacija:=true;n:=n+1;
    end;
    if izraz[n]='(' then
    begin
      t:=n;br_zagrada:=1;n:=n+1;izraz1[0]:=chr(0);
      while (izraz[n]<>')')or(br_zagrada<>1) do
      begin
        if izraz[n]='(' then br_zagrada:=br_zagrada+1;
        if izraz[n]=')' then br_zagrada:=br_zagrada-1;
        izraz1[n-t]:=izraz[n];n:=n+1;
        izraz1[0]:=chr(ord(izraz1[0])+1)
      end;
      n:=n+1;
      calculate(izraz1,br[br_br])
    end
    else
    begin
      t:=1;
      while (s[t].name<>izraz[n]) do
        t:=t+1;
      for m:=1 to br_vre do
        br[br_br,m]:=s[t].value[m];
      n:=n+1;
    end;
    if negacija then
    begin
      for m:=1 to br_vre do
        if br[br_br,m] then br[br_br,m]:=false else
br[br_br,m]:=true;
      negacija:=false
    end;
    if n<length(izraz) then
    begin
      op[br_br]:=izraz[n];                                   {Vadi
operatore}
      case izraz[n] of
        '+','*':n:=n+1;
        '=':n:=n+2;
        '<':n:=n+3;
      end;
      br_br:=br_br+1;
    end
  end;
  {Sada jos ide obrada cije ce rezultat smeste u rez!}
  {Prvo se sredjuju * i +}
  n:=1;
  while n<br_br do
  begin
    if op[n] in ['+','*'] then
    begin
      for m:=1 to br_vre do
        if op[n]='+' then br[n,m]:=br[n,m] or br[n+1,m]
        else br[n,m]:=br[n,m] and br[n+1,m];
      for m:=n to br_br-2 do
        op[m]:=op[m+1];
      for m:=n+1 to br_br-1 do
        for t:=1 to br_vre do
          br[m,t]:=br[m+1,t];
      br_br:=br_br-1;n:=n-1;
    end;
    n:=n+1;
  end;
  {Sada su ciste <=> i => gde je prvi br smesten rez!}
  for n:=1 to br_vre do
    rez[n]:=br[1,n];
  for t:=2 to br_br do
  begin
    for n:=1 to br_vre do
      if op[t-1]='<' then rez[n]:=rez[n]=br[t,n]
      else if rez[n] and not(br[t,n]) then rez[n]:=false else
rez[n]:=true
  end;
end;

procedure naslov;
var n:integer;
begin
  ClrScr;
  writeln(izraz);
  for n:=1 to br_name do
    write(s[n].name,chr(186));
  writeln;
  for n:=1 to br_name do
    write(chr(205),chr(206));
  for n:=br_name*2+1 to 79 do
    write(chr(205));
  writeln;
end;

procedure setup;
var m,n,t:integer;
begin
  for n:=1 to maxslo do
  begin
    s[n].name:=' ';
    for m:=1 to maxvre do
    begin
      s[n].value[m]:=false;
      rez[m]:=false;
    end
  end;
  n:=0;m:=0;t:=0;br_name:=0;br_vre:=0;
end;

{************************** Glavni program ****************************}
begin
  ClrScr;
  writeln('          Analizator Bulovih izraza              by
Celenkovic Slobodan 1988 ');
  writeln;
  write('Unesi simbol za tacno <1>: ');
  repeat
    tacno:=ReadKey
  until tacno<>' ';
  writeln(tacno);
  write(' i za netacno <0>: ');
  repeat
    netacno:=ReadKey
  until netacno<>' ';
  writeln(netacno);writeln;
  repeat
    writeln;
    setup;
    writeln('Unesi izraz (* i,+ ili # negacija komplement <=>
ekvivalentno,=> predikat)');
    repeat
      readln(izraz);
    until sintaksa(izraz);
    {Vadi osnovna slova!}
    br_name:=0;
    for n:=1 to length(izraz) do
      if (ord(izraz[n])>=97)and(ord(izraz[n])<=122) then
      begin
        nepostoji:=true;
        for m:=1 to br_name do
          if s[m].name=izraz[n] then nepostoji:=false;
        if nepostoji then
        begin
          br_name:=br_name+1;
          s[br_name].name:=izraz[n];
        end;
      end;
    br_vre:=1;
    for n:=1 to br_name do
      br_vre:=br_vre*2;
    poc_vrednosti(br_vre);
    calculate(izraz,rez);
    tautologija:=true;
    naslov;
    t:=0;
    for n:=1 to br_vre do
    begin
      t:=t+1;
      for m:=1 to br_name do
      begin
        if s[m].value[n] then write(tacno) else write(netacno);
        write(chr(186))
      end;
      for m:=1 to trunc((79-br_name*2)/2) do
        write(' ');
      if rez[n] then write(tacno) else write(netacno);
      writeln(' (',rez[n],')');
      if not(rez[n]) then tautologija:=false;
      if t=21 then
      begin
        t:=0;
        repeat until ReadKey<>'';
        naslov;
      end;
    end;
    if tautologija then writeln('Ovaj izraz je tautologija!');
    write('Jos (d/n)? ');
    odgovor:=citaj;
  until odgovor='N';
end.
```

element stringa (**a[0]**) sadrži znak čiji aski kod određuje broj znakova koje sadrži string (dužinu stringa))

— inače se promenljiva pronalazi u (globalnom) nizu **a** i njene vrednosti se prebacuju u podniz **niza br**

— ispituje se da li je varijabla **negacija true** i ako jeste invertuju se vrednosti **podniza** niza **br** (**true** postaje **false** i obratno)

Pre nego što prebacimo **operaciju** u niz **op**, moramo da ispitamo da li smo završili sa obradom iskaza. Zašto? Zato što je u svakom iskazu uvek poslednji „broj“ a ne operacija. Sa **case** naredbom ispitujemo da li se radi o operaciji dužine jednog (+ *), dva (=>) ili tri (<=>) znaka.

U drugom delu procedura se izvršavaju operacija po prioritetu * (**and**) i + (**or**) (negacija je još u prvom delu procedure izvršena). Pri tom se koristimo ekvivalentnim operacijama **and** ili **or**, a posle svakog izvršavanja bilo koje operacije nizovi **br** i **op** se šiftuju za jedan ulevo i brojač **br_br** se smanjuje za jedan.

Na kraju nam ostaju samo operacije => i <=>, pa se u sledećem prolazu izvršavaju sve preostale operacije (po redu) iz niza **op** i učestvuju svi brojevi iz niza **br**. Zato je prvi „broj“ prebačen u niz **raz** koji sadrži rezultat, uveden brojač **t** koji označava koja se operacija izvršava (**op[t − 1]**) iz niza **op** i koji još „broj“ (**br[t]**) argument osim **rez**-a. Tako u petlji samo izvršavamo operaciju (**t − 1**) između **raz** i **br[t]**.

Ostali deo programa uz pomoć procedure **naslov** ispisuje tablicu sa vrednostima za promenljive i odgovarajuće rezultate iskaza. Zbog specifičnosti seta karaktera raznih kompjutera potrebno je napraviti određene izmene ako se program ne izvršava na IBM PC XT/AT kompatibilcu

Gužva na steku

Program je napisan na „Turbo Pascalu 4.0“ (vrlo lako se prilagođava ranijim verzijama). Pre kompajliranja programa potrebno je memoriju rezervisanu za stek (memorija rezervisana za podatke procedura i funkcija) maksimalno povećati (birate opcije sledećim redom OPTIONS, COMPILER, MEMORY SIZES, STACK SIZE i upisati maksimalnu veličinu steka 65520). To je potrebno zato što se za svaki stepen zagrade procedura **calculate** rekurzivno poziva i „jede“ dosta memorije (sa ovakvim vrednostima za konstante možete koristiti 3 stepena zagrada)

Balansirajući sa konstantama **maxvre, maxsla** (program) **max_br** (procedura **calculate**) možete postići karakteristike koje vam trebaju. Recimo pošto će te retko koristiti više od 4—5 promenljivih slobodno možete smanjiti maxsla na 6 i maxvre na 64 i time dobijete mogućnost korišćenja više stepena zagrada.

Ovaj program je i lep primer rekurzije bez koje bi bio mnogo komplikovaniji. Ako ne verujete, pokušajte da ga prevedete na bejzik.

Slobodan Čelenković

Mali veliki robot

Motori predstavljaju muskulaturu svakog robota. To su njegove „ruke" i „noge", bez kojih se ne bi mogla ostvariti ni jedna kretnja predviđena programom. Motorima upravlja računar, šaljući robotu precizna instrukcije za izvršenje pojedinih pokreta. U ovom nastavku serije o malim kućnim robotima govorićemo o problemima koji se javljaju prilikom upravljanja.

U najširem smislu reči motor je uređaj koji je u stanju da ostvari kontrolisano mehaničko kretanje

Precizno upravljanje motorima predstavlja opšti problem robotike Svi modeli robota (amaterski ili komercijalni) bez izuzetka koriste jednosmerne (DC) ili tzv step (koračne) motore kojima se relativno lako može upravljati

Kod malih kućnih robota motor je obično najkritičniji deo hardvera sa stanovišta upravljanja iz prostog razloga što mali robot nema kompleksne senzore i mehaniku profesionalnog uređaja

Primenu motora kod malih kućnih robota uvek sagledavamo u funkciji obavljanja radnog zadatka robota Zato ćemo izlaganja usmeriti na korišćenje motora u praktičnim situacijama uz objašnjenja fizičkih zakonitosti koje pri tome važe

Jednosmerni motori

Gradnja svakog robota predstavlja proces planiranja projektovanja realizacije i testiranja nakon čega projektant postaje samo korisnik Kao korisniku malog robota na raspolaganju su vam samo *digitalne komande* Fizičke osobine motora i njegovog kontrolera uočavate kao niz loših realizacija u kretanju robota. Ako pri testiranju utvrdite da izvršavanje programskih komandi robota nije jednoznačno to znači da [k]orisnik praktično nije u stanju da upravlja tim [s]istemom Zato pri projektovanju komandi za upravljanje robotom **moramo uzeti u obzir osobine motora,** jer one direktno utiču na realizaciju zadate komande a time i na vršenje rada

U principu realizacija svakog programskog upravljanja jeste *startovanje* ili *kočenje* motora kao **izvršioca** mehaničkog kretanja

Pod MOTOROM kao **izvršnim organom mehaničkog kretanja** podrazumevamo motore svih tipova i vrsta U našem slučaju to su *mali* (DC) jednosmerni motori U daljem tekstu termin **DC motor** uvek će se odnositi na jednosmerni motor male snage (do 500 mW) Za sve druge tipove motora (step naizmenični hidraulični) to će biti striktno naznačeno

Kontroler motora je hardverski modul koji digitalne komande računara transformiše u signal određenog naponskog nivoa i trajanja

Digitalna komanda sa računara definiše kretanje malog robota a pređeni put je proporcionalan vremenu napajanja motora t_1 trajanju komandnog signala Komandni signal napajanja traje t_1 sekundi i za to vreme robot se kreće jer su motori u pogonu

Upravo *trajanje upravljačkog signala* predstavlja komandu upravljanja Ostale fizičke veličine od značaja ili sami usvajamo ili su već određene usvojenim tipom motora (vidi sl 1)

Stani, kreni ...

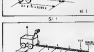
Prvi važan kriterijum upravljanja malim robotom odnosno motorima jeste *zaustavljanje* robota na zadatim koordinatama (pozicioniranje)

Pozicioniranja predstavlja meru tačnosti zaustavljanja robota na zadatim koordinatama (poziciji)

DC motori uvek loše pozicioniraju što znači da mali robot može i da podbaci i da prebaci



Sl 1



Sl 2

predviđeni položaj u rasponu maksimalne greške

Precizno pozicioniranje robota je od izuzetne važnosti jer izbor daljih komandi zavisi isključivo od pozicije na kojoj se robot nalazi Uz srednje veliku grešku pozicioniranja posle par uzastopnih skretanja i vožnja apsolutna greška pozicije se akumulira i dešava se da nismo ni blizu predviđene pozicije (sl 2)

U stvari mi nikad ne znamo tačno poziciju na kojoj će mali robot stati već samo *zonu zaustavljanja* (sl 3) Ako je zona relativno uska ili nam to nije od značaja za vršenje rada smatramo da motor dobro' pozicionira Ako to nije zadovoljavajuće tada pre formiranja komandi upravljanja moramo kompenzovati osobine motora hardverskim sklopovima za pozicioniranja U protivnom pri programiranju moramo uzeti u obzir grešku i zadovoljiti se tako postignutom preciznošću robota

U praksi kod malih kućnih robota najčešće se zadovoljimo zatečenim osobinama motora Upravljanja baziramo na korekciji pozicije upotrebom *senzora* Tako rešen mali robot postaje spor puno šeta napred nazad ali ipak na kraju obavi posao To je klasično projektno rešenje većine komercijalnih kućnih robota

Motori su nevini

Ako ste upravo pomislili da su motori krivi za greške u radu robota — pogrešili ste! Motori veoma precizno obrađuju komande (koliko energije toliko kretanja) i komanda upravljanja su sa svoje strane precizno zadate To su brojevi generisani programom u računaru koje kontroler motora precizno pretvara u izvršne komande upravljanja — signale napajanja motora

Gde je onda greška?

Očigledno ili neko greši ili postoje značajni elementi upravljanja koje nismo uzeli u obzir!

Odgovor na to pitanje glasi nismo u obzir uzeli **masu i inerciju sistema**

Kada na računaru džojstikom zadajete komandu skretanja vaš čovečuljak sasvim sigurno skrene i ne pomišljajte da bi se moglo desiti da se naredba ne izvrši

U robotici kod upravljanja motorima dešava se upravo obrnuto Motor u kretanju predstavlja sistem opterećen dejstvom sila i momenata mehaničkog kretanja Fizika motora zahteva određeni energetski tretman uskladen sa trenutnim opterećenjem U protivnom komanda neće biti izvršena korektno

Izuzetno je teško predvideti upravljanje koje će ispuniti taj tretman pri kočenju i startovanju motora jer je ceo problem *nelinearan.*

Kad se umeša inercija

Da bismo objasnili karakter mehaničkog sistema poslužićemo se analogijom sa računarskim igrama Pretvorićemo već spomenuti uslovno neinertni džojstik u intertni mehanički sistem

Zamislite da je palica džojstika teška gvozdena šipka mase 20 kg Za rukovanje bi bio potreban priličan napor Brz i kratak pokret kao sa standardnom (lakom palicom) ne bismo mogli da napravimo Čovečuljak na ekranu bio bi spor Svaki pokret palicom zahteva ulaganje određene količina energije Ako bismo uložili manje energije nikad ne bismo pomerili palicu na željenu poziciju a čovečuljak na ekranu pomakao bi se samo toliko koliko smo zaista pomerili palicu

Koliko energije je uloženo — toliko se palica pomerila a skretanja na ekranu uvek je proporcionalno realnom pomeranju palice Sa takvom palicom očito nemamo šanse da napravimo male i brze pomeraje

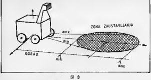
O dobrim refleksima ni govora Promena smera kretanja palice zahteva da *kočenjem* neutrališemo prethodno kretanje a zatim da uložimo energiju za pokretanje palice u novom smeru Kada palica krene u novom smeru za dalje kretanje (zbog inercije) nije potrebna ista količina energije Ako nastavimo da upravljamo istom energijom palica će početi da ubrzava Tada moramo preventivno da kočimo da bismo sprečili prebačaj pozicije

Doslovce se stalno postavlja pitanje u kom momentu u kom pravcu i kolikom silom treba upravljati?

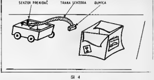
Ljudska ruka kada pokreće palicu džojstika za upravljanje koristi osetila pritiska (dodir) kojim oseća (meri) veličinu sile na palici Taj osećaj predstavlja kontinualno informisanje o trenutnom stanju sila u sistemu Time je omogućeno igraču da pravovremeno i kontinualno reaguje (upravlja) **Jedino stalnim merenjem trenutnih sila i kretanja u sistemu možemo odrediti kada i kolikom silom treba reagovati (upravljati) da bi sistem precizno izvršilo predviđenu komandu**

Motor je upravo takav fizički inercijalni sistem sa vlastitim pravilima igre koja se ne mogu ignorisati

Teorijski je moguće matematičkim jednačinama dosta precizno opisati model DC motora i zatim analizirati njegov odziv na upravljanje s ciljem da se procene nelinearni efekti kočenja



Sl 3



Sl 4

Tu mogućnost treba odmah da zaboravite Od toga se i profesionalcima diže kosa na glavi (ako je imaju)

Odgovor na pitanje ko greši pri upravljanju motorom treba da glasi greši onaj ko zadaje komandu jer nije u stanju da prati promenljive parametre mehaničkog kretenja u sistemu

Pošto nema jednostavnog i jeftinog načina da se te promene mere ni jedan mali robot to ne radi To ne spada u amaterske konstrukcije a i profesionalni uređaji to rade samo uslovno

Jedna opasna greška

Upravljanje *konstantnom silom* je jedino što nam preostaje čime svesno previmo opisanu grešku To je klasičan pristup upravljanju amaterskim konstrukcijama a glavni izvor problema i grešaka u praksi je *prelazni režim motora* u momentu kočenja odnosno startovanja

I pored svih nedostataka roboti sa DC motorima sasvim pristojno rade U kritičnim slučajevima kada greška ugrožava izvršenje zadatka koristimo STEP motor koji *precizno pozicionira* ali drastično ograničava brzinu i snagu robota

Srećom zahtevi upravljanja malim robotom nisu profesionalni (sve dok zadatak kojeg mali robot treba da obavi po preciznosti upravljanja ne postane „profesionalan")

Pošto neke motore na koncu moramo ugraditi oni postaju deo robota i više se ne razmatraju kao zaseban faktor već se praktičnim testiranjem utvrđuju perfomanse robota kao celine To je jedini način da se bez dubljeg teorijskog razmatranja u toku projektovanja predvide praktične mogućnosti budućeg robota

Za preliminarno testiranje motora nije neophodno prethodno sagraditi robota Možete upotrebiti bilo koja kolica" ili staru igračku pod uslovom da ima točkove na koje možete montirati pogon sa motora kojeg želite da testirate

Ako imate teškoća a imaćete ih pogledajte detalje na sl 7 pokušajte da ih iskoristite Snalažljivost je važna osobina graditelja robota

Senzorski — prekidač koga popularno zovu „pipak" ili „pipalica" nije teško napraviti pomoću običnog prekidača i čelične ili plastične trake koja se koristi za okivanje sanduka što nikako ne znači da ne možete upotrebiti bilo šta drugo Za konkretnu realizaciju senzora pogledajte detalj na sl 5

Testiranje motora

Zahtevi koje pred motorni pogon postavlja upravljanje odnose se na pozicioniranje tj startovanje kočenje i prilaz odnosno zaustavljanje pred preprekom

Testiranjem *približno procenjujemo* fizičke i radne karakteristike motora

— pogonsku snagu

— brzinu kretanja i

— inercioni momant motora pri kočenju i startovanju

Proračun nosećih sila (ukupnog tereta) i potrebne pogonske snage motora dosta je zametan posao koji zahteva solidno poznavanje mehanike i fizike Mnogo je jednostavnije eksperimentalno utvrditi maksimalnu težinu koju motorni pogon može pokretati određenom brzinom To je uslov kojeg kasnije u projektu ne smete prekoračiti

Zatim ručno izvodimo predviđene manevre (ili delove manevara) da bismo se uverili u njihovu izvodivost Tako stičemo preliminarno saznanje i iskustvo o mogućnostima upravljanja To se uglavnom odnosi na uočavanje trenutaka *kada je* potrebno kočiti odnosno startovati motorni pogon da bi manevar bio pravilno izvršen Krajnji cilj testa je potvrda manevarskih mogućnosti konkretnih motora pod zadatim opterećenjem

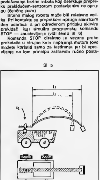
Na slici 4 prikazan je primer testiranja i podešavanja brzine robota koji detektuje prepreku prekidačem-senzorom postavljenim na oprugu (čelično pero)

Brzina malog robota može biti relativno velika Pri kontaktu sa preprekom opruga amortizira deo udaraca a pri određenom pritisku aktivira prekidač koji simulira programsku komandu STOP — zaustavljanja (vidi šemu sl 6)

Komanda STOP direktno je vezana preko prekidača u strujno kolo napajanja motora (ovo možete koristiti samo za testiranje jer bi upravljanje na tom principu zahtevalo ručno postavljanje prekidača u početne položaje nakon svakog kontakta MKR sa preprekom) Prekidač S1 služi za ručno startovanje i pri tome prekidač S2 mora biti zatvoren!

Pošto zbog inercije mali robot ne može trenutno da stane udaljenost robota od prepreke u momentu zaustavljanja zavisi od elastičnosti opruge i količine kretenja (brzine i ukupne mase) Upravo su to fizičke veličine koje moramo proceniti na samom početku projekta

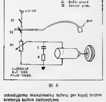
Stavite čvršći prekidač kao senzor prepreke na malog robota (prema sl 5) zatim postavite neučvršćenu prepreku (kartonsku kutiju) Brzinu kretanja malog robota regulišete potenciometrom P1 za kontrolu napona napajanja pogonskih motora



Sl 5

Robotom kroz zid

Za početak utvrdite neku brzinu kretanja u slobodnoj vožnji robota pod predviđenim ukupnim opterećenjem (200—500 grama) Opteretite kolica manjom knjigom ili utegom tako da eventualno možete dodati ili oduzeti deo tereta u toku eksperimentisanja Teret treba učvrstiti običnom gumicom tako da možete testirati klizanje tereta pri kočenju Dodavanjem tereta



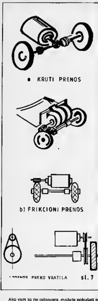
Sl 6

određujemo maksimalnu težinu pri kojoj brzina kretanja kolica zadovoljava

Preopterećenje motora uočavamo kao usporeno kretanje i sporo startovanje iz mira Za maksimalni teret usvojite težinu manju od polovine tereta koju motor *nije mogao da pokrene*

Performanse zaustavljanja (kočenja) najefikasnije procenjujemo eksperimentom jer tada su svi fizički faktori od uticaja na kretanje integrisani u silu udarca o prepreku i grešku zaustavljanja

Sada opterećenog robota puštate da udari u prepreku nekom manjom početnom brzinom U početku ga startujete u blizini prepreke da ne dobije puni zalet Vodite računa da savijanje opruge u kontaktu sa preprekom bude manje od 40% njene maksimalne dužine (vidi sl 4 i 5) Ako brzina i kočenje nisu prilagođeni mali robot će odgurati ili oštetiti prepreku što znači da treba smanjiti brzinu



Ako vam to ne odgovara, možete pokušati sa dužom oprugom. Tada ranije otkrivamo prepreku i ranije aktiviramo STOP komandu, čime se dobija duži put kočenja. Ako ni tada nemamo zadovoljavajuće kočenje u prilazu prepreci, jedino što nam preostaje jeste da još smanjimo brzinu ili da umanjimo teret i probamo ponovo. Kada se uverimo da neće doći do mehaničkog loma, u kutiju stavljamo teret u vidu manje knjige i na isti način testiramo zaustavljanje pred preprekom.

Ovakvo eksperimentalno procenjivanje karakteristika motornog pogona ima izuzetno edukativan karakter bez obzira na moguće kvarove, uz napomenu da ovako jednostavan trik (kočenje oprugom) nije moguće uvek primeniti.

Ostalo je još dosta stvari vezanih za upravljanje motorima o kojima ovde nismo govorili. To će biti izloženo uz konkretan hardverski modul malog kućnog robota u sledećim nastavcima serije.

Branislav Bingulac

Hakerska priča

Code name: SSO

Ogroman umni potencijal domaćih hakera troši se uglavnom na razbijanje zaštita, ubacivanje poruka i ostavljanje sopstvenih imena u inostrane programe (sa naročitim osvrtom na brojeve telefona). Zato evo jednog scenarija koji će, nadamo se, podstaći neki skriveni domaći programerski talenat da napiše novu, dinamičnu igru. Ta nova, dinamična i preko mere uzbudljiva igra može svom tvorcu da donese lep profit na domaćem tržištu (koliko deset kutija žvake) ako ne doživi sudbinu inostranih programa (pročitaj nekoliko redova iznad ovog). U edukativnost programa nećemo se mnogo upuštati — sve će vam biti jasno čim sednete za kompjuter.

U igri upravljate sprajtom nezdravo žute boje koji predstavlja srednjoškolca užarenih izbuljenih očiju (koje se kreiraju efektom FLASH) i savitljive kičme. Ova poslednja osobina dobro dođe i na početku, ali će se naročito korisnom pokazati na višim nivoima.

Prvi ekran predstavlja učionicu u kojoj se bira novo omladinsko rukovodstvo škole. Vaš zadatak je da što je moguće duže trčite kroz učionicu i nosite ogromnu tablu na kojoj piše „Glasaj za mene". Pri tome morate vešto da preskačete noge koje vam vaši drugovi neprestano podmeću. Ukoliko vam se desi da se u toku 20 minuta nijednom ne sapletete, vaši drugovi će se najednom odmaći od vas kad da ste kužni. Tada se pojavljuje veliki sprajt koji predstavlja profesora zaduženog za rad sa omladinskom organizacijom u srednjoj školi. Lagano kliznuvši preko ekrana (pomoću najbolje SCROLL rutine) profesorski sprajt XOR-ovaće se sa vašim. Tog momenta lik učenika kojim upravljate promeniće boju od nezdravo žute u crvenu. U delu ekrana koji predstavlja vaš status u toku igre zasvetleće slova „POČETNIK".

Sledeći ekran predstavlja vašu mesnu zajednicu. Cilj vam je da u svaki ulaz svake zgrade zalepite plakat sa obaveštenjem o sastanku omladinaca vaše M. Z. Plakat je ispisan ćirilicom, što za programera igre predstavlja dodatni napor kod definisanja karaktera. Elem, što se igre tiče, da ni ovaj zadatak ne ispadne suviše lak, stara se gomila neprijateljskih likova. Treba samo pomenuti pse lutalice koji, bar se vama tako čini, naviru niotkuda, a zapravo prebivaju u odbačenim kartonskim kutijama rasutim po celoj mesnoj zajednici. Ove pse naročito izaziva crvena boja vašeg sprajta. Njih ćete najlakše oterati preciznim udarcem u gubicu.

Tako ometani jednom, borite se protiv druge napasti, to su mnogobrojna dečurlija koja sa naročitim uživanjem cepaju plakate koje ste na jedvite jade zalepili. Pojedinačni primerak dečurlije možete srediti nežnim ali snažnim udarcem debelog svežnja plakata koji nosite. Udarac u ruku donosi 5 poena, pogodak u leđa 10, dok za udarac u glavu dobijate bonus od 50 poena. Deca su vraški pokretljiva i borba protiv njih je naročito teška.

Da bi cela stvar bila još komplikovanija, pri svakom udarcu ubojna moć svežnja sve se više smanjuje pa neprestano morate trčati u mesnu zajednicu po nove plakate. Tamo možete da se malo odmorite i povratite energiju, ali se ne smete zadržati suviše dugo, jer počinjete naglo da gubite bodove. Nivo se smatra pređenim kada uspete da zalepite 50 plakata ili da istučete toliko dečurlije da osvojite 10000 poena. Napominjemo da ovaj ekran nije tako težak kao što na prvi pogled izgleda, ali zahteva uvežbavanje od najmanje nedelju dana.

Kada uspešno izvršite postavljeni zadatak, na ekranu se pojavljuje treptajuća poruka „SPREMI SE ZA SLEDEĆI NIVO" na svim jezicima naroda i narodnosti, a vi dobijate status „STARI AKTIVISTA".

Treći nivo je pravi odmor u poređenju sa drugim. Područje po kome se krećete je vaša tiha opština koja, zbog vašeg prisustva, neće još dugo ostati takva. Pošto ste, zahvaljujući svojim refleksima i dobrom džojstiku glatko prošli drugi ekran, kao stari aktivista imate novi zadatak: treba da sakupite određeni broj stanovnika vaše opštine (do 16 do 25 godine) i da njima popunite opštinsku radnu brigadu za lokalnu radnu akciju. Ekranom stalno protrčavaju različite muške i ženske figure, a vi treba da u mnoštvu istih prepoznate određeni uzrast i da ga povedete sa sobom. Jedna od boljih strategija je da prvo pokupite nekoliko vreća sa novcem razbacanih po celom ekranu i da više puta protrčite tamo-amo. To će namamiti likove koji predstavljaju učenike i studente da vas slede, a vi ih morate što brže odvesti do mesta obeleženog onim slovom X. Pri tome vas pratioci gađaju najrazličitijim predmetima, kao što su prazne pivske boce, truo paradajz, pikado-strelice i školske torbe (one su naročito opasne po život zbog velike težine). Ukoliko ne izgubite suviše životne energije, dobijate po jedan poen za svaku dovedenu osobu. Vremenski limit ne postoji, sve zavisi od toga koliko energije imate. Kada na ovom nivou sakupite 50 bodova (toliko treba da sakupite za radnu brigadu), stičete status „ISKUSAN".

Kada uspešno pređete i treći nivo, vaš lik se pojavljuje u nečemu što podseća na bioskop, ali tek malo kasnije prepoznajete salu za sastanke u Gradskoj konferenciji omladine. Zadatak je da se izvučete što tiše i neprimetnije. Morate se pažljivo provući između redova stolica i po kojeg usnulog delegata (kompjuter sintetiše trokanalno hrkanje). Na RND određenom mestu stolica će zaškripati i to će izazvati buđenje prisutnih. Oni će početi da vas gađaju ogromnim količinama materijala („geštetner-literature"), rasutim svuda unaokolo. Ovde treba da dođe do izražaja vaša umešnost u rukovanju džojstikom i sposobnost u krivudanju ekranom. Ako vas pogodi samo jedan „štos" materijala, kompjuter vas baca na početak nivoa.

U slučaju da uspešno pređete i ovaj nivo (u šta iskreno sumnjamo), vaš rejting postaje „SMRTONOSAN". Sada imate sve predispozicije da se nađete na petom (i za mnoge igrače poslednjem) nivou ove nadasve dinamične igre. Prvo na ekranu za trenutak blesne tajanstvena šifra „RKSSO", zatim se pojavljuje vaš sprajt crven do neprepoznatljivosti. Zadatak je suviše jednostavan: treba što duže ostati u igri! Ovde morate da izbegavate neke čudne prilike sa magnetofonima i mikrofonima u rukama, koje vas neprestano vijaju i množe se geometrijskom progresijom. Ako naletite na neki od mikrofona ili se zapetljate u neki kabel (koji ove prilike takođe vuku za sobom), vraćate se na prethodni nivo.

Zatim morate da izbegavate velike cipele koje ponekad protrče preko ekrana. U slučaju da vas nagaze, gubite bodove i vaša energija se drastično smanjuje, ali ne ginete. Međutim, najopasniji su veliki zelenkasti likovi koji se često pojavljuju na ekranu i strahovito su brzi. Nasumice trče u svim pravcima, no, sva sreća, nisu mnogo inteligentni. Ako vas neki od njih zakači, vraćate se na početak igre. Posle slučajno odabranog vremenskog intervala ekran se iznenada zamrači, pa se naglo osvetli i na njemu se pojavljuju blešteća slova: „CONGRATULATIONS! YOU WON THE HIGHEST PLACE IN THE SSO! YOU ARE FIRST AMONG THE LAST AND THE MASTER OF THIS SOCIETY!" I tako igra se završava, a vi ostajete sedeći uz kompjuter, slomljenog džojstika ali punog srca.

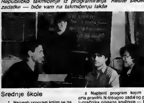
Zoltan Čala

378

Pet plus

Zadaci za takmičenja

U organizaciji Društva matematičara SR Srbija održaće se Republičko takmičenje iz programiranja. Rešite sledeće zadatke — biće vam na takmičenju lakše



Srednje škole

1 Napisati program kojim se na osnovu N različitih rezultata trkača na 100 m datih nizom A(1) A(2) ... A(N) (indeks odgovara startnom broju) formira niz C(1) C(2) ... C(N) gde je vrednost C(I) indeks takmičara koji se plasirao na I-to mesto

2 Nizom koordinata (X(I) Y(I)) I=1,2 ... N data su lokacije N objekata u novosagrađenom naselju. Svaki objekt ima prizemne prostorije predviđene za neki od pratećih objekata (samoposluga, dom zdravlja, butik ...) Napisati program kojim se određuju objekti (njihov indeks) koji su najpovoljniji za otvaranje doma zdravlja tj. čiji je zbir rastojanja od ostalih objekata najmanji

3 Data je matrica Anxn čiji su elementi

1 ako postoji direktan jednosmeran put od grada I do A(I J)=

do grada J

O, u suprotnom

Napisati program kojim se matrica Anxn transformiše u matricu iz koje se vidi koji su gradovi u vezi bez obzira na broj presedanja

4 Napisati program kojim se ispituje da li postoji broj oblika $\overline{AAA}$ koji se može razložiti na sumu kvadrata prirodnih brojeva koji obrazuju aritmetičku progresiju

5 Napisati program kojim se štampaju sve particije prirodnog broja N. Particija broja N su oni različiti prirodni brojevi čiji je zbir jednak N

6 Napisati program kojim se određuje indeks početka i kraja maksimalnog periodičnog odsečka (sa više od 3 elementa) niza A(1), A(2) ... A(n) uzimajući u obzir samo one odsečke u kojima se perioda sadrži ceo broj puta. Npr za odsečak [k 1] se kaže da je periodičan ako $(\exists p)(\forall i \in [k\ 1])(p \mid 1-k+1 \wedge I+p \leq 1 => A(I)=A(I+p))$

7 Napisati program kojim se crta pravilni šesnaestougao sa svim dijagonalama osim onih koje prolaze kroz centar

8 Napisati program kojim se crta pravilni N-trougao zadatog poluprečnika opisane kružnice — R i centra — (XC YC) i K njemu homotetičnih likova na osnovu zadatog centra homotetije (XH YH) i koeficijenta homotetije H

Osnovne škole

1 Napisati program koji određuje razbijanje dinarskog iznosa od Xdin na minimalan broj novčića od 100din 20din 5din i 1din

2 Napisati program kojim se štampaju svi trocifreni brojevi $\overline{ABC}$ koji imaju svojstvo

$\overline{ABC}=(\overline{AB})^2-\overline{C}^2$ gde su nepoznate cifre

$0 < A \leq 9 \quad 0 \leq B \leq 8 \text{ i } 0 \leq C \leq 9$

(npr $147=14^2-7^2$)

3 Prodavnica ima pakovanje deterdženta u džakovima od 16kg 17kg i 21kg. Napisati program koji će odrediti kako da se kupcu isporuči naručenih 185kg bez otvaranja (presipanja iz) džakova

4 U nizu 19885138 ... svaka cifra počev od pete jednaka je poslednjoj cifri od suma četiri prethodne cifre. Počev od koje cifre će se ponovo naići na početnu kombinaciju 1988?

5 Tri prijatelja su bili očevici saobraćajnog prekršaja čiji je počinilac pobegao. Broj automobila (četvorocifren) niko od njih nije zapamtio. Međutim jedan je primetio da je taj broj deljiv sa 2 sa 7 i sa 11, drugi je primetio da u zapisu broja učestvuju samo dve cifre a treći je primetio da je suma cifara broja jednaka 30. Napisati program koji na osnovu podataka svedoka određuje registarski broj automobila

6 Napisati program kojim se generiše slučajan ceo broj na intervalu [100 999], a zatim mu se cifre ispremeštaju tako da se dobije trocifren broj $\overline{ABC}$ čije cifre zadovoljavaju poredak A<B>C. Štampati generisani i novoformirani broj

Pripremio
Milan Čabarkapa

Uređuje: Nevenka Spalević

Arhimedesov kutak

Najbolji rešavači

Prvulović Miloš i ovom prilikom od „Arhimedesa" dobija knjigu na dar, a biblioteka Matematičke gimnazije „Veljko Vlahović" iz Beograda iz koje je prispelo najviše rešenja dobija još jedan ukoričen komplet „Računara".

Najbolji rešavač februarskih konkursnih zadataka, Krašnjak Dubravko iz Kutine, takođe dobija knjigu, a Paljetak Nikola i Mazić Boris osvojili su u tom kolu za svoju školu MIOC Zadar ukoričen komplet „Računara" za 1987 godinu.

Osim što u prošlom broju nismo objavili imena osvajača nagrada, napravili smo još jedan propust. Predrag Mišković, učenik drugog razreda iz Bora, omaškom je izostavljen sa spiska najboljih rešavača. Njegovi programi ocenjeni su sa 7 bodova, pa je prema tome on treći na februarskoj listi najboljih mladih programera. Izvinjavamo se Predragu i čitaocima.

Kvalifikacije su gotove i finalisti godišnjeg takmičenja uskoro će biti poznati. Deset najboljih rešavača pozvaćemo prve nedelje maja i obavestiti ih o detaljima završnog takmičenja. Ako ste nestrpljivi i ne možete da sačekate junski broj „Računara" u kome će sve ove informacije biti objavljene, slušajte radio emisiju „Čip i sedam jarića" prvog programa Radio Beograda subotom u dva ili nazovite redakciju „Računara". Sva prispela rešenja zadataka moćiće da se vide u prostorijama redakcije tokom prve nedelje maja. Komisija će uvažiti

	POENI								
TAKMIČAR	1	2	3	4	5	6	7	8	UKUPNO
ANDREK ROLAND (7), SOMBOR	0	0	2	0	0	0	0	0	2
ANDRIĆ DAVOR (8) ŽIVINICE	1	2	0	0	1	1	3	2	10
BALIĆ VLATKO (8) SPLIT	3	2	0	0	3	1	3	0	12
BANDA VLADO (8) ZAGREB	0	0	0	0	3	2	3	3	11
BEGOVIĆ EDIN (1) B POLJE	0	0	0	0	0	0	3	3	6
BEROŠ IVO (2) JELSA	3	1	2	3	1	2	4	4	20
BOTIĆ ESVED (2) VOGOŠĆA	1	2	0	0	0	0	0	0	3
BRKIĆ SVEMIR (8) BIJELJINA	1	3	0	0	0	0	0	0	4
BULATOVIĆ NIKOLA (8) N BEOGRAD	0	0	0	0	2	2	0	0	4
BUTUROVIĆ ŽELJKA (1) BEOGRAD	0	0	0	0	0	0	4	3	7
GLUŠICA SINIŠA (1) KARLOVAC	0	0	0	0	0	0	3	3	6
GRUJIĆ JELENA (2) SUBOTICA	2	2	0	0	0	0	0	0	4
IKODINOVIĆ IGOR (1) BEOGRAD	2	3	3	3	3	3	3	4	24
JOVANOVIĆ ALEKSANDAR (1) B BAŠTA	0	0	0	0	0	0	3	2	5
JOVIĆIĆ ALEKSANDAR (2) KRUŠEVAC	0	0	0	0	3	0	4	3	10
LAZIĆ RANKO (6) BEOGRAD	0	0	0	0	3	3	3	4	13
KOVAČEVIĆ BRANKO (2) CRVENKA	3	3	2	0	3	1	4	4	20
KRAŠNJAK DUBRAVKO (2) KUTINA	0	0	0	0	4	4	3	4	15
KRMAR ALEKSANDAR (1) RIJEKA	0	0	0	0	0	0	3	4	7
KUČERA ŽELJKO (2) B LUKA	0	0	0	0	0	0	4	1	5
LUKAČEVIĆ SINIŠA (1), VUKOVAR	2	3	0	0	1	2	3	4	15
MAVRETIĆ ZVONIMIR (1) OSIJEK	0	0	0	0	0	0	3	3	6
MAZIĆ BORIS (2) ZADAR	0	0	2	2	4	3	3	1	15
MALINOVIĆ SINIŠA (8) SUBOTICA	2	2	0	0	0	0	0	0	4
MANČIĆ NEBOJŠA (2) NIŠ	2	2	0	0	0	0	0	0	4
MATKOVIĆ ZORAN (1) V KLADUŠA	0	0	0	0	0	0	2	2	4
MESTRIĆ DAVORIN (1) ZAGREB	0	0	0	0	0	0	3	3	6
MILINKOVIĆ VLASTIMIR (2), N BEOGRAD	0	0	0	0	0	0	4	3	7
MITEVSKI DEJAN (2) SKOPJE	0	0	0	0	4	1	4	3	12
MIŠKOVIĆ PREDRAG (2) BOR	2	3	2	0	4	3	3	3	20
NOVELJIĆ NENAD (2) SPLIT	0	0	0	0	0	0	3	3	6
OBRADOVIĆ DAVOR (8) SPLIT	2	2	4	2	0	0	3	3	16
ORMAN NEDŽAD (2) ZENICA	2	3	0	0	0	0	3	0	8
PAVLOVIĆ DEJAN (2) MLADENOVAC	0	0	0	0	4	3	4	3	14
PALJETAK NIKOLA (2) ZADAR	4	3	0	0	4	3	5	3	22
PRVULOVIĆ MILOŠ (1) BEOGRAD	4	5	4	4	0	0	4	5	26
RAKIĆ IGOR (2) S MITROVICA	0	0	0	0	4	3	3	4	14
RAKONCA JOSIP (2) SPLIT	0	0	0	0	0	0	3	1	4
RANČIĆ VLADIMIR (8) PIROT	0	0	0	0	0	0	3	3	6
SMILJANIĆ DRAGAN (2) GOSPIĆ	3	4	0	0	4	2	0	0	13
STAMENKOVIĆ IVAN (1) BEOGRAD	0	0	0	0	3	3	3	4	13
STANISAVLJEVIĆ IVAN (7) ŠID	5	3	3	4	4	3	2	3	27
TRIFUNOVIĆ ALEKSANDAR (2) BEOGRAD	1	2	3	3	0	0	0	0	9
HRNJICA NEDŽAD (8) SARAJEVO	0	0	0	0	2	3	0	0	5
CVETKOVIĆ IVAN (8) POČEKOVINA	2	0	0	0	0	0	0	0	2
CVETKOVIĆ SINIŠA (8) SPLIT	2	1	2	0	3	1	3	1	13
ĆUSIĆ DEJAN (8) VRHOVINE	0	0	0	0	0	0	4	3	7
ŠLJIVAR SLAVEN (8) ZADAR	0	0	0	0	0	0	2	3	5
ŠĆEKIĆ NIKOLA (2) BEOGRAD	0	0	0	0	0	0	3	3	6
ŠĆEPANOVIĆ BOJAN (2) BEOGRAD	2	3	3	0	3	4	4	4	23

sve osnovane primedbe. Radi boljeg uvida u trenutni plasman objavljujemo kompletan spisak takmičara i dosada osvojenih poena. Ako rešenja nekog zadatka nisu prispela, ocenjena su sa 0 poena.

Ne zaboravite da su moguće i bitne promene ove liste jer su u aprilskom broju objavljeni i dopunski zadaci. Poeni koji se dobiju za njihova rešenja zamenjuju najslabije bodovana rešenja. Tako, ako bi, na primer Lukačević Siniša dobio na dopunskim zadacima 2,3,3,4 i 4 poena, oni bi zamenili bodove koje je dobio za 3, 4, 5, 6 i 1. zadatak pa bi umesto 15, koliko sada ima imao ukupno 24 poena i izbio na treće mesto u generalnom plasmanu. Ako bi pak Prvulović Miloš dobio iste poene, bili bi promenjeni samo poeni za 5. i 6. zadatak. Stanisavljević Ivan bi istim poenima svoj skor mogao popraviti samo za tri boda. Čak i neko ko dosada uopšte nije učestvovao u takmičenju, mogao bi, zahvaljujući ovim dopunskim zadacima, doći pravo na treće mesto rang liste, ali to je ipak samo teoretska mogućnost.

Na kraju, zahvaljujemo svim takmičarima na učešću i trudu i nadamo se da ćemo se družiti i sledeće školske godine.

Komisija u sastavu Milan Čabarkapa, Gorica Gargelj i Nevenka Spalević, svi profesori programiranja Matematičke gimnazije u Beogradu, i ovom prilikom pregledala je i ocenila rešenja konkursnih zadataka prispela u redakciju do zaključenja broja.

Prema tim ocenama sastavljena je martovska rang lista najboljih jugoslovenskih mladih programera.

Najbolji rešavači martovskih konkursnih zadataka

1	PRVULOVIĆ MILOŠ (1), BEOGRAD	9
2	BEROŠ IVO (2), JELSA	8
3	KOVAČEVIĆ BRANKO (2), CRVENKA	8
4	PALJETAK NIKOLA (2), ZADAR	8
5	ŠĆEPANOVIĆ BOJAN (2), BEOGRAD	8
6.	BUTUROVIĆ ŽELJKA (1), BEOGRAD	7
7	IKODINOVIĆ IGOR (1), BEOGRAD	7
8	JOVIČIĆ ALEKSANDAR (2), KRUŠEVAC	7
9	LAZAR RANKO (6), BEOGRAD	7
10	KRAŠNJAK DUBRAVKO (2), KUTINA	7
11	KRMAR ALEKSANDAR (1), RIJEKA	7
12	LUKAČEVIĆ SINIŠA (1), VUKOVAR	7
13	MILINKOVIĆ VLASTIMIR (2), N. BEOGRAD	7
14	MITEVSKI DEJAN (2), SKOPJE	7
15	PAVLOVIĆ DEJAN (2), MLADENOVAC	7
16.	RAKIĆ IGOR (2), S. MITROVICA	7
17	STAMENKOVIĆ IVAN (1), BEOGRAD	7
18	ČUŠIĆ DEJAN (8), VRHOVINE	7

U generalnom plasmanu trenutno su najbolji

1	STANISAVLJEVIĆ IVAN (7), ŠID	27
2	PRVULOVIĆ MILOŠ (1), BEOGRAD	26
3	IKODINOVIĆ IGOR (1), BEOGRAD	24
4	ŠĆEPANOVIĆ BOJAN (2), BEOGRAD	23
5	PALJETAK NIKOLA (2), ZADAR	22

Na osnovu pristiglih rešenja za februarske konkursne zadatke, izdvojena su sledeća rešenja.

Za prvi zadatak izabrana su tri rešenja, od kojih su prva dva elegantna po ideji, ali bez mogućnosti uopštavanja (u slučaju da se promeni interval i priroda brojeva), a treće nema eleganciju prve dva, ali ima mogućnost uopštavanja.

Rešenje 1 (Igora Rakića uz neke male dodatke)

```
10 DIM B(39)
20 FOR X=1 TO 1000
30 INPUT "BROJ";A
40 IF A<1 OR A>39 THEN GO TO 30
50 B(A)=B(A)+1
60 NEXT X
70 FOR X=1 TO 39:B(X)=B(X)+ X /100:NEXT X
80 FOR P=1 TO 38
90 FOR Q=P+1 TO 39
100 IF B(P)< B(Q) THEN B=B(P):B(P)=B(Q):B(Q)=B
110 NEXT Q
120 NEXT P
130   FOR   X=1  TO  39:PRINT   "BROJ  ";(B(X)-INT(B(X)))*100;" SE
POJAVLJUJE ";INT(B(X));" PUTA":NEXT X
140 END
```

Rešenje 2 (Igora Ikodinovića)

```
10 DIM C(39)
20 FOR I=1 TO 1000
30 INPUT I
40 IF I> 39 OR Y< 1 THEN GO TO 30
50 C(I)=C(I)+1
60 NEXT I
70 FOR L=1 TO 39
80 MAX= 0
90 FOR K=1 TO 39
100 IF C(K) > MAX THEN MAX=C(K):S=K
110 NEXT K
120 IF MAX >0 THEN PRINT S; C(S)=0:S=0:GO TO 140
130 STOP
140 NEXT L
```

Rešenje 3 (Ivana Stamenkovića)

```
10 INPUT N:      : A(N),B(N),C(N)
20 FOR I=1      N: INPUT A(I): NEXT I
30 FOR X=1 )    N-1: FOR Y=1 TO N
40 IF A(X) >    (I) THEN SWAP A(X),A(I)
50 NEXT I :NE   X
60 C=A(1): K=1
70 FOR I=1 TO N
80 IF A(I)=C THEN 100
90 B(K)=A(I-1): K=K+1: C=A(I)
100 C(K)=C(K)+1
110 NEXT I
120 B(K)=A(I-1)
130 IF K=1 THEN GO TO 170
140 FOR X=1 TO K-1: FOR I=X+1 TO K
150 IF C(X) >C(I) THEN SWAP C(X),C(I): SWAP B(X),B(I)
160 NEXT I :NEXT Y
170 FOR I=1 TO K: PRINT B(I);" SE PONAVLJA ";C(I);" PUTA"  :NEXT I
180 END
```

(Napomena: naredba swap x, y možemo zameniti sa p=x: x=y:y=p)

Rešenje drugog zadatka su po ideji većinom slična: prvo se zadati niz sortirao u rastućem (opadajućem) poretku, a onda se vršila zamena mesta članovima sortiranog niza. Primer jednog takvog rešenja je i sledeće rešenje (Ivana Stamenkovića).

```
10 INPUT N: DIM A(N+1)
20 FOR I=1 TO N: INPUT I(I): NEXT
30 FOR X=1 TO N-1: FOR Y=X+1 TO N
40 IF A(X) >A(Y) THEN SWAP A(X),A(I)
50 NEXT I:NEXT X
60 X=0
70 FOR I=1 TO INT((N-1)/2): X=X+2
80 FOR J=N+1 TO X+1 STEP -1: A(J)=A(J-1)
90 NEXT J:A(X)=A(N+1): NEXT I
100 FOR J=1 TO N: PRINT A(I): NEXT I
110 END
```

Pored ovakvog načina rešavanja, moglo se i jednostavnije, direktnim uređenjem zadatog niza u testerasti poredak. Jedno takvo rešenje je i sledeće (M. Čabarkape).

```
10 'TESTERASTO UREDJENJE NIZA
20 INPUT"BR. ELEMENATA NIZA=";N
30 DIM A(N)
40 PRINT "UNESI ELEMENTE NIZA"
50 FOR I=1 TO N
60   PRINT"A(";I;")=";
70   INPUT A(I)
80 NEXT I
90 P=-1
100 FOR I=1 TO N-1
110   P=-P
120   IF P*(A(I)-A(I+1))<0 THEN 160
130   Q=A(I)
140   A(I)=A(I+1)
150   A(I+1)=Q
160 NEXT I
170 PRINT "TESTERASTO UREDJEN NIZ JE:"
180 FOR I=1 TO N
190   PRINT "A(";I;")=";A(I)
200 NEXT I
210 END
```

Konkursni zadaci za mesec mart su našim rešavačima, sudeći po velikom broju prispelih rešenja, bili laki ali mi ćemo opet izdvojiti i prikazati samo neka reprezentativna rešenja.

Prvi zadatak, po prirodi vrlo lak, rešili su svi učesnici na sličan način, a mi dajemo rešenje Nikole Paljetka, koji se vrlo lepo osvrnuo na problem tačnosti računanja (na SPECTRUMU).

```
10 INPUT "UNESI KOORDINATE CENTRA I RADIJUS KRUGA ";A,B,R
20 INPUT "UNESI KOORDINATE TACKE T ";X,Y
30 X1=A-X: Y1=B-Y
40 P=-(STR$ (SQR(X1*X1+Y1*Y1)))
50 PRINT "TACKA T( ";X;",";Y;")";
60 IF VAL(P$)<= R THEN "PRIPADA ";:GO TO 80
70 PRINT "NE PRIPADA"
80 PRINT "KRUGU S CENTROM U TACKI (";A;",";B;") I S RADIJUSOM ";R
90 END
```

Za drugi zadatak stiglo je dosta rešenja sa puno različitih ideja. Izdvajamo dva rešenja (od kojih je prvo, u sličnim verzijama, bilo dosta zastupljeno među rešenjima) za koja je karakteristična jednostavnost i elegancija ideje a i mogućnost generalizacije.

Rešenje 1 (Krašnjak Dubravka uz malu dopunu za veću tačnost)

```
10 DEF FNP(A,B,C,D,E,F)=ABS((A*(D-F)+C*(F-B)+E*(B-D))/2
20 INPUT "UNESITE KOORDINATE TEMENA TROUGLA: ";X1;Y1;X2;Y2;X3;Y3
30 INPUT "UNESITE KOORDINATE TACKE: ";A;B
40 PT=FNP(X1,Y1,X2,Y2,X3,Y3)
50 PP=FNP(X1,Y1,X2,Y2,A,B)+FNP(X2,Y2,X3,Y3,A,B)+
   + FNP(X3,Y3,A,B,X1,Y1)
60 IF ABS(PT-PP) < 0.000001 THEN PRINT "TACKA SE NALAZI UNUTAR
   TROUGLA ":STOP
70 PRINT "TACKA SE NALAZI IZVAN TROUGLA"
80 END
```

Rešenje 2 (Prvulović Miloša)

```
10 DIM X(5),Y(5)
20 INPUT "KOORDINATE TACKE T X,Y ";X,Y
30 FOR A=1 TO 3
40 PRINT "KOORDINATE ";A;" TEMENA":INPUT X(A),Y(A)
50 NEXT
60 FOR A=1 TO 3
70 X(A+3)=X(A): Y(A+3)=Y(A)
80 NEXT
90 FOR A=1 TO 3
100 IF X(A)=X(A+1) THEN DEF FNZNAK(X,Y)=SGN(X-X(A)): GO TO 120
110 DEF FNZNAK(X,Y)=SGN((Y(A)-Y(A+1))*(X-X(A))/(X(A)-X(A+1))+
    +Y(A)-Y)
120 IF FNZNAK(A+2),Y(A+2))=0 THEN PRINT "TEMENA TROUGLA SU
    KOLINEARNA,UNESI IH PONOVO":GO TO 30
130 IF FNZNAK(X,Y)=0 OR  FNZNAK(X,Y)=FNZNAK(X(A+2),Y(A+2))  THEN
    NEXT ELSE GO TO 150
140 PRINT "TACKA PRIPADA TROUGLU ":END
150 PRINT "TACKA NE PRIPADA TROUGLU ":END
(Napomena : program je pisan na BASIC-u racunara CPC464 )
```

Podaci u banci

Na našim stranicama redovno možete naći informacije o Programima za numerička izračunavanja, ali o takozvanoj poslovnoj obradi koja predstavlja najveću oblast primene računara samo s vremena na vreme i pomalo. Tekst nove saradnice mr Lidija Popović o osnovnim pojmovima obrade podataka, koji objavljujemo u ovom broju, samo je uvod u seriju o primeni računara u poslovnoj obradi koju na našim stranicama možete pratiti od jeseni. Kako mr Lidija Popović, pored toga što se bavi uvođenjem računarskog poslovanja u radne organizacije, dobar deo vremena provodi i u školi jer je njeno osnovno zanimanje profesor, u ovoj seriji tekstova posebnu pažnju posvetiće primeni računara u obrazovnim organizacijama

Kreiranje svakog ozbiljnijeg programa iz oblasti obrade podataka se sastoji iz dva dela

1 ORGANIZACIJE PODATAKA
2 IZRADE PROGRAMA

Upoznajte se na samom početku sa osnovnim elementima i postupcima koji čine ova delove

Organizacija podataka

Organizacija podataka je hijerarhijska i čine je

POLJE—>SLOG—>DATOTEKA—>BAZA PODATAKA

Osnovni elementi u organizaciji podatka je POLJE engleski termin FIELD. Polje predstavlja jedinstven podatak koji se može opisati svojim karakteristikama kao što su vrsta podatka, dužina i slično

SLOG engleski termin RECORD jeste niz polja koja čine logičnu celinu

DATOTEKA engleski termin FILE jeste niz slogova logično povezanih u celinu

U ovom primeru SLOG je kombinacija polja IME I PREZIME DA-

Primer

• IME I PREZIME • DATUM ROĐENJA • mesto ROĐENJA • GRAD • ULICA • BROJ
• A D R E S A

TUM ROĐENJA MESTO ROĐENJA i adrese koju čine polja GRAD ULICA BROJ i predstavlja logičnu celinu koju čine lični podaci jedne osobe

Ako formiramo niz slogova koji će sadržati ova podatka za određenu grupu ljudi na primer radnike jedne organizacije ili stanovnike jedne opštine dobićemo datoteku ličnih podataka pomenute grupe ljudi

Najčešće ima više datoteka s

Primer

STANIĆ VERA	16/01/86	BEOGRAD	ZEMUN	M. TITA	13	
STEVIĆ ANA	23/10/46	ZAGREB	ZEMUN	GETEOVA	34	

podacima koje obrađujemo. Zbog toga se mora definisati i najviši stepen u organizaciji podataka BAZA PODATAKA engleski termin DATA BASE

BAZA PODATAKA je skup DATOTEKA u kojima se čuvaju razni podaci. Ove datoteke mogu biti povezane ili nepovezane što zavisi od osnovnih principa koji se usvajaju na početku rada sa bazom podataka. Sledeći primer ilustruje bazu podataka sa povezanim datotekama

Primer

U obradi ličnih dohodaka BAZA PODATAKA se sastoji od sledećih povezanih datoteka.

— LIČNI PODACI — lični podaci radnika

— SATNINA — podaci potrebni za obračun ličnih dohodaka

— IZOSTANCI — podaci o bolovanjima vojnim vežbama i drugim izostancima

— ODBICI — podaci o odbicima kreditima obaveznim doprinosima i slično

Organizacija podataka sastoji se od sledećih postupaka

1 Analiza podataka i njihovih međusobnih veza

2 Formiranje strukture slogova i osmišljavanja datoteka i baze podataka

3 Kreiranje datoteka koje čine bazu podataka korišćenjem odgovarajućih komandi dibejz (dBASE) jezika

Izrada programa

PROGRAM je niz instrukcija na osnovu kojih računar izvodi niz određenih aktivnosti. Instrukcije i komanda koje čine program su engleska reči koja računar ne razume stoga postoje programi koji prevode te komanda i instrukcije na jezik računara. Takve programe zovemo PREVODIOCI ili INTERPRETERI. Postoji veliki broj specijalnih programa koje nazivamo programski jezici namenjenih izradi programa iz oblasti obrade podataka (Kobol, PL1, dabejz itd). Kada su u pitanju personalni računari najčešće se koristi dibejz. Stoga će primeri koji slede biti vezani upravo za ovaj programski jezik

Primer

Ako u programu date komandu CREATE za kreiranje datoteke dibejz III interpreter će najpre uporediti reč CREATE sa spiskom komandi koje ima. Ako pronađe komandu u tom spisku izvršiće aktivnosti vezane za tu instrukciju u suprotnom javiće korisniku da je napravio grešku

Pisanje programa

Prvo pitanje koje sebi postavljate je „kako da napišem program i sačuvam ga u memoriji računara ? Taj postupak se obavlja prema određenim pravilima sa kojima ćemo se sada upoznali

Pisanje programa omogućava tekst procesor koji je ugrađen u dibejz III plus jezik. Procesor se aktivira komandom MODIFY COMMAND iza koja se navodi ime programa koji želite da napišete. Na primer komandom MODIFY COMMAND PRIMER aktiviraće se tekst procesor za pisanje programa PRIMER. Isti je postupak i u slučaju kada već postoji program PRIMER a želite da ga ispravite

Kad je postupak pisanja programa završen potrebno je sačuvati program, za šta se takođe izdaju odgovarajuće komande ovog tekst procesora. Rezultat tog postupka je da se u memoriji računara program nalazi sačuvan pod imenom sledećeg oblika

PRIMER PRG

Reč PRG koja sledi iza tačke i imena PRIMER služi za identifikaciju sadržaja koji je označen imenom PRIMER, a koji u ovom slučaju označava program

Sledeći postupak je da startujete program koristeći komandu DO. Rezultat tog postupka biće izvršavanje svih aktivnosti koja ste programom predvideli

Program možete pisati i pomoću ma kog drugog tekst procesora, a postoji i mogućnost da taj drugi tekst procesor takođe startujete komandom MODIFY COMMAND. To se postiže definisanjem odgovarajućeg parametra u KONFIGURACIONOJ DATOTECI

Konfiguraciona datoteka

Konfiguraciona datoteka CONFIG DB služi za definisanje osnovnih parametara bitnih za rad u dibejz III plus jeziku. Ona se takođe kreira komandom MODIFY COMMAND CONFIG DB. Pogledajte primer jedne konfiguracione datoteke

```
SAVETY=OFF
TEDIT=EDIT EXE
F5=SET ECHO OFF
F6=DISPLAY STRUCTURE
F7=DISPLAY STATUS
```

Ovom konfiguracionom datotekom definisani su sledeći parametri

— SAFETY parametar svojim statusom ON ili OFF uslovljava postojanje ili ukidanje dvostruke provere prepisivanja preko postojeće datoteke. U ovom slučaju ta provera je ukinuta

— TEDIT parametar daje mogućnost definisanja teksl procesora koji želite da koristite umesto već predviđenog. U ovom slučaju to je EDIT EXE

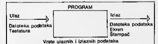
— F5, F6, F7 parametri definišu programske dirke na tastaturi računara, tako da se prilikom na određenu dirku izvrši konkretna dibejz III plus komanda

Ulazni i izlazni podaci

Svaki program ima ulazne i izlazne podatke. Ovi termini se odnose respektivno, na podatke koje program treba da obradi i podatke koji nastaju kao rezultat te obrade. Ulazni podaci na primer mogu biti polja ili slogovi datoteka koja se koristi u programu ili podaci koji se unose preko tastature u toku izvršavanja programa. Izlazni podaci mogu biti takođe datoteke, zatim razni izveštaji na ekranu ili štampaču

Jedan od osnovnih problema koja program mora da reši jeste definisanja ekrana za unos ulaznih podataka i definisanja izlaznih ekrana i izveštaja na štampaču

Kada se govori o ulaznim i izlaznim podacima uvek treba imati na umu veoma bitnu činjenicu a to je da kvalitet ulaznih podataka direktno uslovljava kvalitet izlaznih podataka

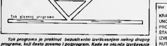


Vrste ulaznih i izlaznih podataka

Tok programa

Tok programa je redosled kojim se izvršavaju komanda i instrukcija u programu. Najjednostavniji programi izvršavaju se pravolinijski odnosno komanda se izvršavaju onim redom kojim su i napisane. Kod iole složenijeg programa to nije moguće već se moraju obezbediti postupci koji omogućuju programu da bezuslovno izvršava postupke da izvršava postupke kada je ispunjen neki uslov i da ponavlja određena postupke dok je zadovoljan neki uslov. Te postupke respektivno označavamo terminima BEZUSLOVNO IZVRŠAVANJE USLOVNO IZVRŠAVANJE I IZVRŠAVANJE U PETLJI. U dibejz III plusu postoje komanda koja omogućuju realizaciju ovih postupaka ▶

S njima ćemo se informativno upoznati nešto kasnije

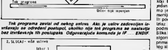


Tok programa je prekinut bezuslovnim izvršavanjem nekog drugog programa, koji često zovemo i potprogram. Kada se okonča izvršavanje tog potprograma, tok programa se nastavlje komandom koja sledi neposredno iza komande kojom je pozvan potprogram. Odgovarajuća komanda dibejza je DO

USLOVNO IZVRŠAVANJE

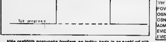


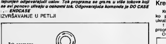
Tok programa zavisi od nekog uslova. Ako je uslov zadovoljen izvršavaju se određeni postupci, ukoliko nije tok programa se nastavlja bez izvršavanja tih postupaka. Odgovarajuća komanda je IF ... ENDIF



Više različitih postupaka izvršava se jedino kada je za svaki od njih ispunjen odgovarajući uslov. Tok programa se grana u više tokova koji se svi ponovo ulivaju u osnovni tok. Odgovarajuća komanda je DO CASE ... ENDCASE

IZVRŠAVANJE U PETLJI





Program izvršava neke postupke sve dok ne bude ispunjen određeni uslov. Kako program ponavlja te postupke sve dok uslov ne bude ispunjen, koristi se i izraz „vrti se u ptelji", pa otuda i ime svog postupka. Kada se uslov ispuni nastavlja se normalni tok programa. Odgovarajuća komanda je DO WHILE ... ENDDO.

Rad sa menijima

Većina složenijih programa koristi rad sa MENIJIMA. Objasnićemo osnovni koncept tog rada. Program startuje sa OSNOVNIM MENIJEM iz koga se mogu birati konkretne aktivnosti. Te aktivnosti takođe mogu biti meniji u tom slučaju to su PODMENIJI osnovnog menija.

Primer koji sledi ilustruje rad sa menijima a odnosi se na program školske evidencije. OSNOVNI MENI imao bi sledeći izgled

Osnovni meni

Ver 3 1	PLM — SOFT	18 33 07
KRAJ RADA		0
UNOS PODATAKA		1
PROMENA PODATAKA		2
ŠTAMPA PODATAKA		3
IZVEŠTAJI		4
RUTINE		5
PROŠIRENJE		6

Osnovne aktivnosti se biraju iz OSNOVNOG MENIJA a u okviru PODMENIJA nabrojani su pojedinačni poslovi koji se izvršavaju u okviru izabrane aktivnosti iz OSNOVNOG MENIJA.

Korišćenje glavnog menija i podmenija ilustruje jedan važan koncept programiranja u dibejzu koji bi se mogao sažeti u izraz jedan ulaz jedan izlaz. Postoji samo jedan put kojim se iz glavnog menija dolazi do željenih aktivnosti ili podmenija i postoji samo jedan izlaz kojim se iz te aktivnosti ili podmenija vraća u glavni meni. Slika koja sledi je šematski prikaz — algoritam prethodno navedenog primera za rad sa menijima. Analiza tog algoritma olakšaće vam shvatanje rada sa menijima i navedenog koncepta.

Podmeni unos — podataka

Ver 3 1	PLM — SOFT	18 33 07
POVRATAK U OSNOVNI MENI		0
OSNOVNI PODACI O ŠKOLI		1
OSNOVNI PODACI O UČENICIMA		2
ADMINISTRATIVNI PODACI		3
EVIDENCIJA OCENA		4
EVIDENCIJA IZOSTANAKA		5

Kreiranje programa

Kreiranje programa čini nekoliko postupaka koje ćemo sada ukratko objasniti.

1 Detaljna analiza i odluka šta program treba da radi

Ovaj postupak obuhvata analizu problema koji hoćemo programski da rešimo. Što ga bolje obavimo imaćemo manje problema u ostalim postupcima. Iako se čini da je lak ovom postupku treba posvetiti posebnu pažnju.

2 Podela celokupnog problema na manje logične celine

Ova podela predstavlja osnovu glavnog koncepta u programiranju poznatog pod imenom STRUKTURNO PROGRAMIRANJE a sastoji se u podeli problema na logične delove koji se lakše mogu programirati nego ceo problem. Ovi delovi se nazivaju programski moduli. Ovakav koncept rada omogućava da se svaki programski modul posebno analizira programira i testira a zatim se moduli povezuju u jedinstven program koji rešava problem u celini.

3 Pisanje programa za svaki pojedinačni modul

Na osnovu prethodno izvršene podele i analize pojedinačnih modula pišemo programe na dibejz ili plusu za svaki modul.

4 Dokumentacija programa

U toku pisanja programa potrebno je formirati dokumentaciju koja olakšava praćenje i razumevanje programa. Ima više načina na koji se to radi sa kojima ćete se kasnije upoznati.

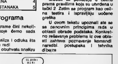
5 Testiranje programa

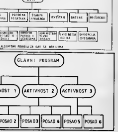
Programski moduli se testiraju pojedinačno jer se na taj način lakše otkrivaju eventualne greške. Testiranjem programa otkrivaju se sve greške i situacije kada neispravno radi. Napomenimo da i pored svih testiranja u programu uvek ostane ponека greška koja se otkrije tek dužim korišćenjem programa u konkretnoj primeni.

6 Formiranje jedinstvenog programa

Moduli se povezuju u celinu prema pravilima koja su utvrđena u tački 2. Zatim se program kao celina testira i ispravljaju uočene greške.

U ovom tekstu upoznali ste se sa osnovnim principima rada u oblasti obrade podataka. Konkretno rešavanje problema iz ove oblasti zahteva poznavanje komandi naredbi postupaka i tehnika dibejza.





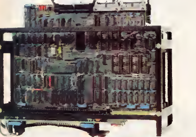
Mr. Lidija Popović

Institut MIHAJLO PUPIN

tim 300

VIŠEPROCESORSKI MIKRORAČUNARSKI SISTEM

Sredнje spregnuti mikroračunarski sistem sa raspodeljenom inteligencijom sledi savremene koncepte povećanja propusne moći replikacojm hardverskih komponenti, prvenstveno procesorskih



modula. Vitalnu vezu među modulima čine MULTIBUS sistemska magistrala i zajednička memorija koja služi kao komunikacioni medij, odnosno centar za poruke.

Rezidentna magistrala procesorskog modula povezuje njegove lokalne resurse, smanjujući tako zasićenje na sistematskoj magistrali. TIM 300 realizuje virtuelnu mašinu kod koje se procesi alocirani na različitim procesorima izvršavaju paralelno, dok programsko okruženje na nivou procesorskog modula može biti jednoprocesno ili višeprocesno.

Heuristički metod raspoređivanja procesa po procesorima minimizira međuprocesorsku komunikaciju i čini vreme razmene poruka zanemarivo malim u poređenju sa ukupnim izvršavanjem procesa.

Zahvaljujući svojoj modularnoj strukturi i alternativnim rešenjima osnovne programske podrške, TIM 300 se lako prilagođava različitim zahtevima u pogledu procesorske snage, memorijskih potreba, nivoa virtuelne mašine itd.



Osnovnu konfiguraciju računara čine:
◇ do 15 računarskih modula sa:
- mikroprocesorom 8086 (do 8MHz) koji ima snažan skup naredbi, širok repertoar načina adresiranja i inherentne predispozicije za multiprocesiranje,
- matematičkim koprocesorom i8087 (opciono),
- dinamičkom RAM memorijom sa 128 ili 512 Kb kome se pristupa bilo sa lokalne magistrale bilo sa Multibusa, postoji mogućnost definisanja veličine dela RAM-a kome se može pristupiti sa Multibusa, kao i njegovo mapiranje u Multibus adresnom prostoru,
- EPROM memorijom do 64 Kb,
- 2 programabilna serijska RS232C kanala na koje se mogu priključiti ergonomski projektovani TIM 001 terminali,
- programabilnim paralelnim kanalom sa 24 U/I linije;
- programabilnim kontrolerom prekida koji omogućuje 8 (sa mogućnošću proširenja do 56) nivoa,
- tri nezavisna programabilna brojača
- mogućnošću produženja lokalne magistrale za sprezanje sa U/I uređajima sa kojima se intenzivno komunicira;

◇ uređaj za upravljanje spoljnim memorijama podržava do dva Winchester



diska različitih kapaciteta, jednu jedinicu disketa bilo 8" bilo 5 1/4" i jednu jedinicu magnetne kasete (streamer)

◇ brza i pouzdana statička RAM memorija od 32Kb predstavlja zajedničku memoriju sistema i zahvaljujući mogućnosti pristupa sa dve strane, omogućuje povezivanje sa drugim TIM 300 ili drugačijim računarom

◇ lokalna magistrala procesora povezuje lokalne resurse i preko adaptera magistrale može se produžiti do 50 m čime je omogućena paralelna komunikacija i sa udaljenim uređajima

Mali oglasi

Cena običnog malog oglasa do deset reči je 6000 dinara Svaka naredna reč košta još 600 dinara

Cena uokvirenog malog oglasa je 8000 dinara i po visinskom centimetru u stupcu širina 9.5 cicera ako oglas nije viši od pet centimetara i 12000 dinara po visinskom centimetru ako je mali oglas visok između pet i deset centimatara

Mali oglas treba dostaviti na adresu redakcije Računari — BIGZ (za male oglase) Bulevar vojvode Mišića 17 11000 Beograd najkasnije do trećeg u mesecu Svi oglasi koji do ovog roka pristignu u redakciju poštom lično i uz određena ograničenja telefonom biće uvršćeni u sledeći broj

Mali oglasi se po pravilu plaćaju unapred bankovnom uplatnicom na račun 60802-603-23264 BIGZ Bulevar vojvoda Mišića 17 11000 Beograd sa obaveznom naznakom Računari mali oglasi Kopiju uplatnice treba obavezno dostaviti zajedno sa tekstom malog oglasa

SPEKTRUM

Veoma povoljno prodajem programe za Spectrum 48K Tel 034/60-373

SPEKTRUMOVCI komplet 2600 dinara Besplaten katalog Veliki izbor tematskih kompleta. Komplet B4 Ballbreaker Indiana Jones Tai Pan Mercenery Ski Simulator Sidewize Batty Eagle's Nest Renegade Telefon 088/33-531

SPEKTRUMOVCI! Komplet 1200 din + kaseta + PTT = 4300 din! Pojedinačno 200 din program Katalog besplatan Imamo sve iz ostalih oglasa METAL SOFT tel 071/645-491

MEPHISTOSOFTWARE kvalitet je privilegija vodećih! Proverite! Tel 024/42--174

COCE SOFTWARE! Sami sastavite komplet — isporuka ODMAH! snimci direktno iz Spectruma. Katalog besplatan Srđan Đurković Bul Revolucije 68 Titograd tel 081/42-546

NOVO — super program za učenje telegrafije Cena = 3000 dinara Kompjuterski rečnici 1 Englesko--srpskohrvatski i obratno (sa dodatkom od 480 kompjuterskih izraza) 2 Nemačko-srpskohrvatski i obratno 3 Englesko nemački i obratno Cena programa = 3000 cena dva programa = 3500 cena tri programa = 4000 Telefon 011/497-662 (od 17 do 19 h)

SPEKTRUMOVCI! Ekskluzivno vam predstavljam nejnovije hitove sa engleskog tržišta Predator Basket Master Rolling Thunder Victory Road The Last Ninja itd Naravno posedujem i sve ostale programe koji su trenutno u Jugoslaviji Spisak je besplatan Zoran Jovanović Cara Uroša 13A/11 18000 Niš. tel 018/61-249

MIKYSOFT vam nudi najnovije igre Sidewalk Garfield Phantom's Club Platoon Captain America Do izlaska još novija igre Tražite katalog Miljenko Borojević Lenjinov trg 1/5 44000 Sisak, tel 044/30-418

Prodajem Centronics interfejs za ZX Spectrum Slobodan Knežević Zagorska 36B Sarajevo tel 071/656-965

COMPUTER SERVICE
VIII Vrbik 33a/6
41000 Zagreb
tel 041/539-277 od 10—12 sati
i od 15—17 sati
— Spectrum Commodora Atari Amstrad
— brzi i kvalitetni popravci
— prodaja joystic-a interface-a, mrežnih ispravljača kablova memorijskih proširenja rezervnih dijelova

Spectrumovci! Velika rasprodaja! Komplet samo 1300 din + kaseta (2300 din) Rok isporuke 1 dan Komplet Z 11 Bride of Franken Vanganza Dizzy Xcecutor Ball Breaker Pitagora ACE 2 Ninja Hamster Challenge of the Gobots Flunky Thunder Komplet Z 12 match Day Gun Ship Driller Combat School Lineker Athena Salamander 720 Nebulus Ryger Monty Game Komplet Z 13 California Games (6 pr) Out Run (2 pr) Grand Prix Simulator Star Ware Lazer Wheels Druid 2 Rubicon Dark Sceptre Komplet Z-14 Super Hang On (4 pr) Pegasus Bridge Heist 2012 SUB Classic Muncher Andy Capp Evening Star Sky Warriors Tuberuba Action Force Tank Level 5 Tamara Vujisić Lenjinova 8/II ulaz 11080 Zemun tel 011/210-334

Spectrumovci! Komplet samo 1300 dinara + kaseta (2300) Sva četiri kompleta samo 4000 din + cena kasete + PTT Rok isporuke 1 dan Najbolje igre 7 Fist 2 Ninja BMX Simulator Scalextrix Kane SF Harrier Leader Board Bomb Jack 2 Eagle's Nest Samurai Feud Najbolje igre 8 Saboteur 2 Army Moves Auf Wied Monty Big Trouble in Little China Transmuter Uchi Mata Enduro Racer Nether Earth Indoor Sports Nemesis-Warlock Spy vs Spy Mario Bros Najbolje igre 9 Barbarian Barbarian 2, F15-Strike Eagle Hydrofool Holywood Poker Flash Gordon Gun Runner GBA Basketball Mag Max Commando 87 Game Over 1 Game Over 2 Najbolje igre 10 Airwolf 2 James Bond -007 Death Wish 3 Falcon Armaggedon Joman, Renegade ATV Simulator Exolon Batty Road Runner (4 pr) Tamara Vujisić, Lenjinova 8/II ulaz 11080 Zemun tel 011/210-334

Veoma povoljno prodajemo programe i igre za Spectrum Kvalitet zagarantovan 011/814-870

SPECTRUM — Prodajem sintetizator ljudskog glasa (10M) dvostruki A/O konvertor (10M) Testirano + softver Tibor Tot tel 021/881-650

SPECTRUMOVCI! Samo odabrani hitovi u 90--min. kompletima 1 komplet = 1500— din kaseta C90 obična = 2500.— Philips = 3800.— PTT = 1000 — Uz komplet dobijate uputstva za svaku igru Pretplatnicima popust 20% Mesečno dva kompleta Kvalitet zagarantovan uverite se K-10 Winter Olimpiad 88 (1—5) Bravestarr Basil Predator (1—4) Mega Apokalipse I Ball Death Ride (1—2) Merlin. Road Wars K 11 Desperado (1—4) Arkham Mannor (1—2) Dan Dare 2 OINK! Double K-12 Antares A.T.F 3D Starfighter Hollyday In Sumaria Katalog besplatan. Clive Shugar 7 Vojvođanske brigade 52 21208 Sr Kamenica

SPECTRUMOVCI! Najnoviji kompleti po ceni od 1000.— din Kaseta C60 obična = 1500 — Philips AGFA = 3000 — PTT = 1000 — din Pretplatnicima 20% popust Uz komplet dobijate uputstvo za svaku igru Isporuka brza i kvalitetna K-60 Collision Course Bravestarr Erik Phantom Deviants, Basil Inside Out. Predator (1—4) Iwo Jima Mega Apocalypse K-61 Desperado (1—4) I Ball 2 Death Ride (1—2) Dan Dare 2 Antares Nihilist Merlin Road Wars (hit) K-62 Arkham Manner (1—2) OINK! Double A.T.F Katalog besplatan EXTRA kvalitet Clive Shugar 7 Vojvođanske brigade 53 21208 Sr Kamenica

ZX TURBO DRIVE (floppy za spectrum) prodajem Turbodrive interface kompletan III dijelova WD 1770 EPROM TTL čipove oscilator štampane ploče konektora flat kabel flopi pogona 360K HW SERVICE PP 96 42300 Čakovec tel 042/54-795

KOMODOR

KOMODORE 16 — novi programi Duško Aleksić Golubinačka, 7a 22320 Inđija tel 022/55-277

C-64/128 tematski kompleti Sexy Duell Komplet (40p)=1600 dinara Katalog Lazo Mitrov Kožuv 1 92400 Strumica tel 0902/24-509

C-64/128 apsolutno najnoviji programi. Komplet (40p)—1600 din Besplatan katalog Lazo Mitrov Kožuv 1 92400 Strumica tel 0902/24-509

C-64/128. super program. Ubacivač (vaših) intro-a za kasetu 1000 dinara Lazo Mitrov Kožuv 1 92400 Strumica tel 0902/24-509

Prodajem komodor 64 sa kasetofonom i palicom i diskete 5.25 i 3.5 inča Tel 011/585-295

KOMODOR 16 116 +4 —10 programa 1400 dinara (STARFORCE FANTATRON TIME TRAK) Siniša Konjević Kaporova 30 78000 Banja Luka tel 078/68-609

C-64 veliki broj uslužnih disk programa i uputstava Besplatan katalog Šinć Ivan Pionirska 16 56000 Vinkovci tel 056/17 583

YU O D — I ovog meseca mnoštvo hit-programa po niskim cenama komplet+PTT+kaseta=3000 dinara Možete se javiti tokom celog meseca Imamo Iznoguda Pro BMX Sim. Dan Dara 2 Tel 037/26--329 (od 20 do 21 30h)

COMMODORE 64 popravci AV-kablovi (dužina proizvoljna) ugradnja reset tipke i kućišta osigurača tel 072/21-057 (9-15)

Commodore 64/128 Fantastična ponuda na jednom telefonu preko 3000 igara (stare najnovije i one za sledeći mesec) Na kaseti ili disku Svi postojeći uslužni programi sa uputstvima pomoć početnicima i naprednijima popravka (otkupljivanje) palica Katalog 750 dinara unapred Miroslav Čakarevski Radoja Domanovića 28/II 11000 Beograd tel 011/417-371

Komodor 64/128 Vrhunski uslužni programi za disk i kasetu Uputstvo za korišćenje disk igre Miroslav Čakarevski Radoja Domanovića 28/II 11000 Beograd tel 011/417-371

C-64 Najidealnija prilika za jeftinu kupovinu 100 najnovijih programa +3 kasete +PTT=8000 dinara Miroslav Čakarevski Radoja Domanovića 28/II 11000 Beograd tel 011/417-371

Komodor 64 Izaberite sami svoj komplet iz ostalih oglasa Kataloge ne šaljem jer posedujem sve programe Miroslav Čakarevski Radoja Domanovića 28/II 11000 Beograd tel 011/417-371

INTERNATIONAL STUDIO — komplet (30 prg)+kaseta+uputstva+PTT=2700 Dva kompleta=5000 Bez intrca K1 Platoon 1—3 Basket M Tetris BMX Kids 2 Droid Dreams K2 Garfield II Predator Besplatan katalog tel 078/68-854

C16 C116 C+4 prodajem programe jeftino Branko Brstina Sabo Mikloša 11 24000 Subotica tel 024/20-096

KRAGUJEVAC CRACKING SERVICE C64/128 — Prvi put u YU — MESEČNI HIT KOMPLET Najbolji programi koji se pojave u toku meseca Hit komplet februara (Platoon 1—3 BMX Kids 2 Predator 1—4 All World G 1—4 Tetris) za nas je već star što garantuje da je K C S pored LCM-a i YU C S -a pouzdan izvor programa Mogućnost PRETPLATE Svaki program VERIFIKUJEMO kvalitet ZAGARANTOVAN OD 5 5 POPUST 20% Besplatan katalog Dragan Prokić Bačvanska 10 34000 Kragujevac.

C-64 Proverite zašto smo bolji od drugih Klub TINA Sombor Bellneka 26 tel 025/23-384

COMMODORE 64 — prevedena literatura, Speed DOS YU-G-ROM 802 EPPROM moduli Dmir Kovačić Trg Pionira 7 41410 Velika Gorica tel 041/719-892

COMMODORE 64—128 — KO1 Platoon 1—3 Gryzor Rastan KO2 Predator Phantom Pro BMX 1—2 KO3 ATV Simulator Stratton Slaine Kompleti sadrže 18—25 programa sa spiskom i programom za štelovanje glave Tematski kompleti Sport Šah Porno Komplet+kaseta+PTT je 4.000 dinara Tel 089/22-178 Denis

Prodajem novi disk 1541C cijena 600 DM tel 056/17-374

Od 1. maja nove cene: oglas do 10 reči 6000 din. — svaka dalja reč 600 din.

KODAKSOFT nudi najnovije programe za Commodore 64/128 Informacije 072/81 1-308 (posilje 14h)

Kupujam disk 1571 za komodor 128 star do 2 godine Dejan Tufegdžić Karađorđeva 169/2 36000 Kraljevo tel 036/334-603

commodore 64
IGRE USLUŽNI I OBRAZOVNI PROGRAMI — sve najnovije cene povoljne katalog besplatan KVALITET ZAGARANTOVAN 100% Sami sastavite katalog ili se oslonite na naš izbor najnovijih igara i kompleta na kasetama i disketama Milan Milašić Beogradski kej 45/63 N Sad tel 021/27 134

C16 116 +4 veliki izbor novih igara Cene povoljne Dragiša Rajković, 3 Oktobar 302/2 19210 Bor tel 030/32-523

Prodajem Commodore 3032 sa zelenim monitorom dva kasetofona i A4 printerom Seikosha GP100A i odličnim građevinskim programima (OKVIRI DINAMIKA RESETKE ROSTILJI FIZIKA ZGRADE i drugo) Basic je kompatibilan sa Commodore 64 Ivica Kozar Omladinska 5A 51000 Rijeka

COMMODORE 20, 16, +4, 64 128 — preko 5000 programa za C-64 Katalog tražite uz markice od 400 din Za VC-20 i C 16 spisak besplatan Šandor Ojerman Rade Končara 23 23000 Zrenjanin

C-64/C-128 KOMPLET 29 Cleverš-Smart Lynx IX Plaloon 1—3 Basket Master Bod Moren Gremlins Snooker Ancient Ta Tetris Knight G II 1—3 XOR# Smashed Callan Tulistrom Pegasus Bridge Blast Impact+ i Ball-2 Battle Walley Energy Warrior Super M Bros Scumball Repel Gruzor+ Thunderforce Tolip Thundercross KOMPLET 30 Renegade Pegasus Bridge 3d L Rescue 1—6 Rigel s Revenge Drak Where T Began Rollaround Werewolves Funitrax B 2 All W Games 1—4 TRAZ Octapolis Giana Sisters Brave Starr Thing Twice-3 Wix Wax Bocce Balls Hel Trick Code Hunter Slaine Enc Viking-2 Stratton Komplet 2000 din dva kompleta 3500 din Besprijekoran kvalitet Ištvan Pakoci M Stanivulovića 92 Zrenjanin tel 023/63-920

HIGHLANOER vam nudi najnovije programe za vaš komodor 64 Komplet 3 Predator 1—4 Moonshadow Frightmare Gyzor Rollaround Rockford Impact 100% XX Bob Moran Champ Sprint Train Robbers Big John Ikari Warrior II Battle Valley Dan Dare II Fortres U C M 40—45 programa+kaseta+PTT=4500 din. Poklon svim naručiocima — program za štimanje glave kasetofona Ako želite da proverite zašto kod nas nema reklamacija — javite se!!! Tel 091/426-890 (Duško) ili 091/426-896 (Šašo) ili Duško Kirovski Bulevar Jana Sandanski 40/6 91000 Skopje

Commodore 64/kasetni programi Basket Master Police Akademy 2 Magnetron 1—8 Card Sharks 1—3 Phantoms Platoon 1—3 Gauntlet 2 Mega Phantom 1—4 Spisak i druge informacije na telefon 031/881-626 Munitlak Vladimir

MI IMAMO SVE najkvalitetnije za disk C64-800 C128 i CP/M 1500 Literatura popusti katalog besplatan Ivančica Kokić I L Ribara 7a 41000 Zagreb 041/573-769

40 odabranih programa kaseta poštarina 4000 din Prvih 10 pokloni Milan tel 071/648-341

Muzički komplet za Commodore 64
Upoznajte sjajne muzičke mogućnosti računara — komponujte na raznim instrumentima Uđite u svijet muzike i uživajte u harmoniji tonova Komplet M/88+nova kaseta+PTT=4000 dinara. Isporuka brza i tačna. Vladimir i Miroslav Nikolić Živka Jošila 9/13 71000 Sarajevo tel 071/648-755

Korisnički programi za Commodore 64 Najbolji korisnički programi grafičke aplikacije reklamni ntiol i asembleri komplet K/88 MS-DOS Emulator Real Writer Democreator 3d Design Wizawrite Geos Easy Script Pascal Forth Profi Asembler Gaph 64 MAE 2 Englian CAD Art Studio 1—2 TT Copy A11 Micro painter Marko Text Simon 's Basic Koala Painter 2 Image System Polfon Diamon 64 sa uputstvom Komplet K/88+nova kaseta+PTT=4000 dinara. Isporuka odmah. Miroslav i Vladimir Nikolić, Živka Jošila 9/13 71000 Sarajevo tel 071/648-755

Šah! Šah! Šah! Za Commodore 64 Jedinstvena prilika za ljubitelje drevne igre Trodimenzionalne figura sjajna grafika detaljna uputstva Komplet S/88+nova kaseta+PTT+uputstvo za svaki šah=4000 dinara. Usluga brza tačna i kvalitetna Miroslav i Vladimir Nikolić Živka Jošila 9/13 71000 Sarajevo tel 071/648-755

Sport! Sport! Sport! Za Commodore 64 Game Set and Match 1—10+Winter Olympic Game 1—10. Najbolje urađeni letnji i zimski sportovi Igrajte i uživajte Tenis fudbal košarka atletika boks, itd Skijaški skokovi slalom bob veleslalom biatlon itd Komplet SPORT-88+nova kaseta+PTT=4000 din Isporuka odmah Usluga kvalitetna i tačna Miroslav i Vladimir Nikolić Živka Jošila 9/13 71000 Sarajevo tel 071/648-755

COMPUTER SERVICE
VII Vibik 33a/6
41000 Zagreb
tel 041/539-277 od 10—12 sati i od 15—17 sati
— Spectrum Commodore Atari Amstrad
— brzi i kvalitetni popravci
— prodaja joystic-a interface-a, mrežnih ispravljača kablova memorijskih proširenja rezervnih dijelova

C-64/128/CPM prodajem uslužne disk programe i disk igre Posebni kompleti kazetnih programa. Besplatni katalog Prodajem diskete 5.25 inča Radovan Fijember Klaićeva 44 Zagreb tel 041/572-356

Vihunski razdelnici za snimanje za dva commodoreova kasetofona (samo 3000) i RESETMODULI (4000) Mikica Milovanović Nemanjina 1/1 36000 Kraljevo tel 036/22-597

EDI SOFT — programi za vaš komodor 64 Tražite katalog Zoran Jakovljević Beogradska 9 26000 Pančevo tel 013/42-680

Izuzetno povoljno — izuzetno povoljno!!!!!! Komplet za Commodore 64 sadrži Predator 1—4 Fiefly Timefighter Ikary Warriors II Bob Moran Dan Dare II Frightmare Warewolves of London Vampir Empire Tiger Hell Calkan Thalgon Blodo Valley 1—3 Rolling Thunder Herobotiks President Campaign 88 Task 3 Isno God (A1 Reshid) Arkanoid New Rockford Moon Shadow Cena (najkvalitetniji hitovi maja+kaseta+PTT) =3500 — Komplet sadrži 30—40 igara Plaćanje pouzećem Kvalitet profesionalni Mikica Milovanović Nemanjina 1/1 36000 Kraljevo tel 036/22-597

LUXURY BOY
COMMODORE 64
CALL:045/85-178
— pozdravlja LCM-a YUCS i TWW-a
— poziva na suradnju sve hakera
— za ovaj mjesec posjeduje najnovije disketne i kazetne programe
— za vas razbija i pravi INTRO programe
— naručite besplatan katalog
LXB-Igor Pal Moslavačka 61 41315 Novoselec

D&D

C-64, uslužni muzički programi i igre Komplet U1-8 Ass/Mon (profi ass.) U2 11 grafičkih (Art Studio I 2), U3-10 programskih jezika (Oxford Pascal) 1 komplet+kaseta=4500 d Prodajemo originalne igre Predator California Games Super Circle Summer Games 2 1 igra+kaseta=2500 d Programe možete naručiti i pojedinačno Katalog besplatan. Denis Abramović D Cesarića 67 41090 Zagreb tel 041/700-447

Prodajem za C-64/128. RESET-modul koji od ostaje nakon resetiranja većine programa (5.000 din) EPROM-modul+RESET više turbo programa u modulu+podešavanje glave (19.000 din) Simon s Basic Extended Basic Monitor Easy Script T-prikljućak za dva kasetofona presnimavanje i zaštićenih programa (8.000 din) NAVLAKA-zaštita od prašine za kompjuter disk pasč (1.500) kasetofon (1.200) palice (1.000) svjetlosno pero PROGRAMI +poštarina Zdenko Šimunić, Kolareva 58 41410 V Gorica tel 041/714-608

najnoviji programi
za commodore 64
M & S SOFT
III bulevar 130/193
11070 N. Beograd
tel. (011) 146-744

C-64 EPROM moduli kablovi programi na disku i kazeti uputstva Dvor Borošak Froudeova 88 41020 Zagreb tel 041/522-508

KOLEKCIONARI! Prodajem 21 originalnu igru za Commodora 64 (firme Epyx Firebird US Gold) u originalnom pakovanju Informacije (samo pismeno) na adresu Lale Krivačević Trg E Kardelja 56 81000 Titograd

Commodore 64 — disketni programi i igre Besplatan katalog sa opisima Teofil Rislić Ž Zrenjanina 27 26000 Pančevo tel 013/31-69

AMSTRAD

Prodajem Schneider CPC 464 sa zelenim monitorom programima i literaturom Tel 035/261-647

NAJNOVIJE — NAJJEFTINIJE za vaš CPC-464 Besplatan katalog SOFTHING SOFTWARE Prečko 41 41000 Zagreb tel 320-569 posllje 16h

AMSTRADOVCI! Super jeftine programe (80 din) nabavite kod Erasofta Besplatan katalog Dragan Biljić Pionirska 1 31000 T Užice tel 031/27-365

AMSTRADOVCI!!! PRILEP SOFT vam nudi najnovije programe komplet 1000 din oko 18 programa pojedinačno 150 din Besplatan katalog Tel 098/25-688 Ordanco ili Rubin Negrioski Nikšićka 6 97500 Prilep

CPC-464! Najbolje uslužne programe snimam na kvalitetne TOK kasete. Ivan C tel 016/43-710

AMSTRAD CPC-464 800 programa komplet 1000 din pojedinačno 180 din Katalog besplatan Slavko Mardeleski M Tita 63/a 22 000 Ruma

AMSOFT YU

AMSOFT YU CP/M Software predstavlja najnovije CP/M programe QUASAR 2-statistički paket SCRIVENER-računanje unutar text procesora LOCOSRIPT 2-Joyce POCKET WORD STAR MGX (Mathematic s & Graphic Extensions) —Joyce DESK TOP PUBLISHER-Joyce Character Designer-Joyce PAGEMAKER DBASE Compiler DBASE Phone Manager DBASE Mail Manager AMSTAT 1 2 3 4 E-BASIC PL/0 Compiler PILOT SUPERCOPY RAM DISC 64 LIBRARY (subdirectory) SQUEEZE UNSQEEZE micro COBOL FORTH-83 Small-C (floating point) NEWCPM TURBO PASCAL ROS 3 3 CBASIC 80 EXBASIC DR DRAW DR GRAPH CP/M igre Strike Force Harier Batman Megar3 Almazar Menopoly Bacarrat Adventure! 3D Clock Chess Mogućnost isporuke svih programa sa YU setom AMSOFT YU Trg Republike 4 41000 Zagreb, tel 041/270-777

Komplet CP/M i utility programa komplet LANGUAGES FORTRAN SMALL C, JRT PASCAL FORTH Komplet TEXT Wordstar Mailmerge Prospell Rotate Komplet STATISTIKA AMSTAT 1—4 Komplet 3 0 dBASE II Superbasic 2 Wordstar 3 34 ZIP SOI Komplet 2 2 Microscript Micropen Microspread Cambase Database Novi AMSDOS programi Masterfile 6128 YU Tasword 6128 YU Tasspell Mini Office 2 Profi Painter Hardware proširenje 464 na 6128 (CP/M 3 0) Silicon disc 256K Lightpen EPROM-programator EPROM-i sa YU setom za printer AMSOFT YU Trg Republike 4 41000 Zagreb tel 041/270-777

AMSTRAD-SCHNEIDER 464/664/6128
Za projektantske ustanove i pojedince programi za proračune
Statički neodređenih sistema sa 80 i više čvorova (ramovi) i rešetki u ravni za 120 i više čvorova
Slobodan Knežević, Zagorska 36B Sarajevo tel 071/414-000 (324) od 18h 656-965

Oglasi u okviru do 2 cm 16000 din. — svaki dalji centimetar 8000 din.

Napravite sami komplet (1800 dinara) Najnoviji kompleti 2000 dinara — TANK COMMANDO ZIGY ALPIN GAMES Siniša Jakobović Trg slobode 22 54400 Đakovo tel 054/842-460

DRACULA SOFT — za vas CPC 464 a od sada i za ORIC NOVU 64 nudi najnovija programe u kompletima i pojedinačno ilustrovani katalog 350 din Jovica Božinović D Slambolića 11 18360 Svrljig

JOYCE dBase II Multiplan Amelet Locoscript 2 Duasai 2 Desktop Publisher DR-Graph itd Bert Parać Laginjina 9 41000 Zagreb tel 041/410-090

MIKISOFT!!! 1800 (komplet) + 2000 (kaseta) + 800 (ptt) = 4800 dinara Katalog besplatan Tomislav Janković D Petrović-Šana 20/2 14220 Lazarevac tel 011/814-604

PRODAJEM povoljno neraspakovan ocarinjen AMSTRAD-SCHNEIDER 6128 sa zelenim monitorom Tomislav Janković Dušana Petrovića 20/2 14220 Lazarevac 011/814-604

MIKISOFT!!! 1800 + kaseta + PTT Širok izbor najnovijih programa Katalog besplatan Tomislav Janković Dušana Petrovića 20/2 14220 Lazarevac 011/814-604

AMSTRAD 6128 664 464 (+DISC)
- Posedujem najveći broj uslužnih programa i igara na disketama u YU
- Za stalne kupce dajem popust
- Besplatan opširan katalog sa dužinama i opisima igara i opisima svih uslužnih programa
- Garantujem najprofesionalniju uslugu u zemlji
Prodajem Schneider CPC 464 + zeleni monitor + joystick + 800 programa + literatura ocarinjen. Cijena 95 miliona starih (ili 950 DM). Blažo Bojić Ante Zuaniće 15A 88000 Mostar tel 088/415-203

Dva džojstika JY-1 za CPC 464 video kasete Tel 011/402-503

ATARI

ATARI XL/XE Programi po cijeni od 200—400 din Katalog besplatan! Pozovite 077/472-289 Dalibor

Prodajem ATARI STM520 SF 354 sa programima Tel 013/44-354

ATARI MASTER CLUB XL/XE Za vaš Atari XL/XE preko 1100 programa na kaseti i disketi 35 knjiga magazini Atari User i Monitor i još mnogo toga u katalogu od 30 strana Za katalog poslati 800 din Slobodan Jovanov Prvomajska 2-A 23000 Zrenjanin

ATARI XL/XE COMPUTERS!!! Programiranje u Atari bejziku uvodi vas u sve tajne Atarijevog bejzika operacionog sistema Srpskohrvatski — originalno izdanje Cijena 9800 din Emsi Husaković Zahirovića 11/A 72000 Zenica tel 072/35-119

ATARI XL/XE — razmjena igara (Druid Trailbalzer ...) poslovnih programa Emli Husaković Zahirovića 11/A 72000 Zenica tel 072/35-118

ATARI XL/XE Druid Loco S Lobsters. Katalog besplatan Saša Halembek Vrsar R Končara 12 tel 053/41-446

ATARI XL programi Za SPECIJAL FOTO-KATALOG (110 fotografija najboljih igara) poslati 1000 din. ili za običan katalog 300 din Ivan Vučurović JNA 65 26210 Kovačica

Atari XL/XE V C S i ovog mjeseca sa najnovijim hitovima za vas Living Daylights Phantom Brew Biz Gunlaw Ballblazer Aztec Microrhithm Miroslav Mikić Đure Salaja 41 42000 Varaždin tel 042/53-977

ATARI XL/XE DON'T PANIC SOFTWARE!!! Mnogo novih programa! Cene 0—300 dinara! Besplatan katalog! Zoran Iliev Širok Dol 25 92000 Štip tel 092/35-843 (Zoki) ili 22-182 (Vane)

ASC Teenager' -Atari Soft Club Teenager' nudi svim vlasnicima ATARI XL/XE računala razmjenu najnovijih programa na kasetama. Ako ste zainteresirani za ovaj vid saradnje pošaljite nam vaš spisak programa s naznačenom adresom Vaše ponude šaljite na adresu Tomislav Vicković Doverska 9/IX 58000 Split

Prodajem Atari 800 XL + kasetofon + 2 palice tel 011/4899-675

HL/XE povoljno programi katalog besplatan Goran Grgić Žrtava fašizma 2 79260 Sanski Most 079/85-610

XL/XE najnoviji programi i kompleti Snimamo na dva kanala DARKVUD SOFT Dane Tešanović Ravno polje 76330 Ugljevik

ATARI ST HARDWARE
— monokormatski monitor SM 124 u sve tri rezolucije
— 720 KB novi tanki floppy disk
— 1 44MB dvostruki floppy disk
— SF 354 kao dvostrani floppy
— epromprogramator (2716—27011)
— video-digitalizator
— sound-digitalizator
— ATARI ST kao osciloskop
— PAL — interface (512 boja na TV)
— ROM-modul ili EPROM-bank
— proširenje memorije
— sve vrste TDS-eva
— SCART-kabel
— diskete 2DD i 1DD
— razmjena Hardware-a
— besplatan katalog
GARANCIJA 12 MJESECI
Tel 042/817-596 od 15—22 h
Brezje 38 42311 Lopatinec

EXCELENT SOFT ST Captain Blood Enduro Racer Defender of the Crown Baker Street Barbarian Program 700 din Pri svakoj narudžbi jedan od ovih programa snimamo besplatno! Moguća i RAZMJENA! Provjerite! Tel 056/15-424

Atari Soft Klub vam nudi svoje profesionalne usluge za modele Atari XL i XE najbolji izbor programa i literature na YU tržištu specijalne popuste Ko ne veruje — neka se uveri! Kvalitet je zagarantovan. Katalog 300 din Dejan Lacmanović Sinđelićeva 31a 23000 Zrenjanin tel 66-879

Nekorišteni garancija P N P electronic 720K disk za ST prodajem NEC drive, Iralo u kućištu Kablovi Tel 041/448-225

*** R. MILJAKOVIĆ ***
ATARI ST !
Besplatan katalog, sa više od 500 programa i preko 100 naslova različite literature
Posebni popusti za 5,10,20,50 programa
Express isporuka, provera svih snimljenih programa, niske cene !!
H. POLJE C.I/48
61260 LJ.-POLJE
tel 061/487-427

YOCHA SOFT — igre i uslužni programi na kasetama za ATARI 800XL/130XE Katalog besplatan Jeftino prodajem dva joystick-a Redostik i jedan Atari joystick Jovan Milović 18205 Niška Banja Trg Republike 34 tel 018/860-582

ATARI XL/XE najpovoljnije u Jugi kod KOKOSOFTA! Programi (1942 Arkanoid) u najjeftinijim kompletima (1 000 — 1 500) Katalog besplatan Domagoj Sekulić V Korajca bb 55330 Velika tel 055/53-685

ATARI ST — veliki izbor najnovijih programa. Najjeftiniji kompleti po vašem izboru Više od 800 programa Diskete 3 5 Katalog besplatan Boris Gruden Turinina 10 41 020 Zagreb tel 041/676-228 i 436-002

IBM

Prodajem računar PC/XT IBM kompatibilan nov Tel 021/321-857

BOR SOFT
IBM PC XT/AT — ATARI ST
— Veliki izbor najnovijih PC i ST programa
— Emulacija programa PC — ST
— Kopiranje na 360K 720K i 1 2 MB
— Kopiranje PC programa sa 5 25 na 3 5 i obrnuto
— Prodaja disketa od 5 25 i 3 5
— Originalna literatura i programska uputstva
— Prevodi na srpskohrvatski latinica
›Stručni saveti za RO pri kupovini hardware-a i software-a
› Izrada software-skih rešenja za RO po želji
Besplatan katalog!
S. Dimitrijević Post Restante 19210 Bor

Mijenjam programe za PC Prodajem nagrađeni originalni Tekući račun Tel 054/24-461

Prodajem i razmjenjujem vrhunske uslužne programe i kompajlere za IBM-PC tel 034/851-551

LITERATURA za PC Using Word Perfect Chi-Writer Turbo Prolog MS Fortran MS C Windows Lotus itd. Dejan tel 011/150-835

Prodajem kompatibilni računar (IBM) model PCM-XT10MHz sa tastaturom i zelenim monitorom. Monohrom 12 model HMM 1210 herkules kartica 640K dva flopi diska i miš. Tel 021/323-729

COLOR TV — PC XT AT Neka vaš TV postane COLOR MONITOR (CGA)! Informacije na tel 062/301-632

VENTURA RASTAVLJA REČI PREMA SRPSKOHRVATSKIM PRAVILIMA
Nudimo vam kao dopunu standardnog Ventura Publisher paketa module za On-Line rastavljanje reči po pravilima srpskohrvatskog jezika Maksimalno iskorišćenje prostora estetsko rastavljanje
Jednostavna instalacija Isporuka ODMAH!
Za sve informacije
Milan Vatimirović, 011/147-535

SOFTVER za IBM PC/XT/AT prodajem i mijenjam Najbolji izbor najnovijih 900 uslužnih programa i 100 igara (PCAD 20D ORACLE 20D MS C v 5 00 9D FONT GEM IV+2D) Snimam programe na diskete 5.25 i 3.5 Tjedno novi programi zato tražite novi besplatan katalog Baksa Zdenko Ivana Milutinovića 34 41040 Zagreb tel 041/254-561

Diskete 5 25 Cijena 2500. Enles 071/214-319

NOVO ZA PC! KOMPLETI! Program + diskete + uputstvo + plastične korice Povoljno za RO Kada kupujete — kupujte kvalitetno! CHARLIE SOFT — 071/628-519

Charlie Soft
IBM PC Programi i literatura.
Diskete 5.25" Povoljno za RO
Otes B-35 ul.5/7 71210 Ilidža
tel: 071/628-519

IBM PC. Prodajem programe jeftino Katalog besplatan Vladimir Šurjić Bratstva i jedinstva 25 71000 Sarajevo

EE IBM PC SOFTWARE
Najveći izbor software-a za IBM PC u Yugoslaviji po najnižim cijenama Double DOS v4 0 In a Vision Vizawrite vl 45 MAce Utility 4 DAN-/CAD/MOVIE 3D MS C + QUICK C v5 0 + upute. i još preko 249000Kb vrhunskih programa najpoznatijih svjetskih proizvođača. Literature! Pokloni! EKSTRA popusti! Katalog besplatan! ROK ISPORUKE 24 SATA! EE SOFTWARE Martićeva 31 78000 Banja Luka tel 078/40-540

TERMIČKI PRORAČUN (prema najnovijem JUS-u) Linijski specifični i ukupni gubici difuzije toplotna stabilnost toplotni otpor vazduha interaktivan rad bez tablica i prethodnih priprema B B Budisavljević, tel 011/642-163

AMIGA

Prodajem Amigu 800 diskete kablova color TV-monitor novo neraspakovano ocarinjeno. Jovan 022/424-428

AMIGA programi! Šandor Đarman R Končara 23 23000 Zrenjanin

Uokvireni oglasi između 6 i 10 cm 12000 din. po centimetru

AMIGA

AMIGA — veliki broj kvalitetnih uslužnih programa i igara koje možete nabaviti po pristupačnoj ceni Cena jednog programa na jednoj disketi je samo 2 500 dinara Sva obaveštenja i naravno besplatan katalog možete dobiti na adresu Mikan Videnović AVNOJ C-2/II 10 19000 Zaječar ili telefon 019/28-800 (15—22h)

HARDVER

Prodajem spektrumov matrični Seikosha GP 50 štampač Tel 051/731-481

Prodajem delove za PC (CPU DMA CRT RAM) i Atari ST (RAM) flopi drajv 5 inca — J Mbajta monitor Ifd Lale od 16—18h, tel 018/339-976

PRODAJEM — potpuno novu disketnu jedinicu (5 25 Inch — Matsushita Communication Ind Co Ltd LR55468 E66486 — DC12V/0 25A DC 5V/04A) za IBM PC i kompatibilna računara (cijena 250DM) — PHILIPS BM7502/00g zelani monitor veoma povoljan za QL i slična kućna računare (cijena 290 DM), — SONY mini kasetofon sa MIC i EAR ulazima regulacijom brzina motora Jlat mikrofonom koji je veoma povoljan za rad sa SPECTRUMOM ali je isto tako i dobar diktafon — čemu je prvenstveno namjenjen (cijena 150 DM) — FRINTER BROTHER M-1009 (Centronics i RS232C interface) sa dodatnim traktorom za perforiran papir (cijena 400 DM) Informacije na tel 062/842-576 od 15 do 20h

Prodajem ATARI 520 STM NEC 1037 može i odvojeno Zoran Butić Nazorova 43B 41000 Zagreb, tel 041/445-938

Prodajem SHARP PC 1403 (Matrix Operation) Danijel Turina Zagrebačka 128/3 41410 Velika Gorica

AMSTRAD CPC-464 sa zelenim monitorom prodajem Nazvati poslije 16h tel 075/236-586

DISKETE 150 komada 5 1/4 prodajem za 28 000 dinara Robert Golob 62212 Sentilj 117/a

SPECTRUM 48K potpuno nov prodajem Tel 041/512-928 posle 15h

Prodajem C-128 i kasetofon ocarinjen Aco tel 078/56-643

Prodajem zelani monitor GT-65 Aco tel 078/56-643

Prodajem računar PC/XT monitor, miš, turbo tel 021/616-143

ATARI 520 STR dvostrani disk diskete 3 5 i 5 25 Boris Gruden Turinina 10 41020 Zagreb tel 041/676-228 i 436-002

Prodajem računar PC/XT IBM kompatibilan, nov Telefon 021/321-857

Prodajem spektrum 48K sa interfejsom I mikrodrajvom INES tastaturom programima i literaturom Momčilo 018/323-202

COMPUTER SERVICE
VIII Vrbik 33a/6
41000 Zagreb
tel 041/539-277 od 10—12 sati i od 15—17 sati
— Spectrum Commodore Atari Amstrad
— brzi i kvalitetni popravci
— prodaja joystic-a interface-a mrežnih ispravljača kablova memorijskih proširenja rezervnih dijelova

Commodore plus 4 neupotrebljavan prodajem Tel 011/460-212

Prodajem Commodore 64C kasetofon i joystick za 50 miliona Damir tel 042/53-259 (do 12 sati)

EVEREX EGA autoswitch 640×480 software ocarinjena garancija — 011/434-480

C-64 disk 1541C i sva prateća oprema Teofil Ristić Ž Zrenjanina 27 26000 Pančevo tel 013/31-68

LITERATURA

Najnovija knjiga za sistemsko programiranje (190 strana) Jasmin Galijašević tel 078/39-126

PROFESIONALNI PREVODI
KOMODOR 64 Priručnik (4 500) Programer's Reference Guide (4 500) Mašinsko programiranje (3 000) Grafika i zvuk (2 200) Matematika (2 200) Disk 1541 (1 800) Uputstva za uslužne programe Simon's Basic Multiplan Praktikalk (po 1 300) Vizawrite Easy Script MAE Help-64+ Paskal Star Graf Supergrafik (po 1 000). U kompletu 23 000.
SPEKTRUM Mašinac za početnike (3 300) Napredni mašinac (3 800) Devpak-3 (1 300) U kompletu 7 100. ROM rutine (knjiga) 8 000.
AMSTRAD/SNAJDER Priručnik CPC 464 (knjiga) (4 000) Locomot v Basic (3 800) Mašinsko programiranje (3 500) Uputstva za uslužne programe Masterfile Devpak Tasword Paskal Multiplan (po 1 300) U kompletu 14 000 Priručnik CPC 6128 (knjiga) (8 000)
KOMPJUTER BIBLIOTEKA Bate Jankovića 75 32000 Čačak tel 032/30-34

RAZNO

Prodajem sve vrste igar Naročajte brezplačan katalog kom=1000 din kaseta=2200 din PTT=600 din Matjaž Gradišnik Mislinjska Dobrava 99 62381 Podgorje tel 062/849-266

SHARP MZ 700/800 dva kompleta izabranih igara Maxican Adventure Schach 700 Advoka Gunman Sportsman Bomber Man Backgammon Le Mans Star Trek Club Golf Branko Štulić Šulekova 23 41000 Zagreb tel 041/448-225

Comax

Tim za izradu programa iz oblasti mašinskih konstrukcija nudi vam programe za proračun geometrije i dinamike zupčanika, proračun zavarenih spojeva i vijčanih veza Informacije i prospektni materijal Predrag Stanojević Čede Zarevca 24 37000 Kruševac tel 037/32-172

AMSTRAD/SPECTRUM — razmjenjujemo i prodajemo programe besplatan katalog Dražen Hajting i Miro Sokčević Trg Pobjede 19 55000 Slavonski Brod tel 055/231-987 (prije podne)

Diskete 3 5 i 5 25 prodajem Boris Gruden Turinina 10 41020 Zagreb tel 041/676-228 i 436-002

Prodajem diskete 3 5 i 5.25 inča po najnižim cenama u zemlji Tel 011/585-295

Prodajem diskete 5 25 i 3 5 DS/DD PC XT hard disk novo ocarinjeno tel 034/714-948

-InfoGen-

„Računari" su nekada pisali o tome Ili je o tome pisao neki od preostalih osam kompjuterskih časopisa Ali

KADA? KAKO? KOLIKO?

potražite odgovore na sva slična pitanja u Indeksu tema računara i autora u knjizi

VODIČ KROZ JUGOSLOVENSKU RAČUNARSKU LITERATURU 1981—87

(Preko 260 strana formata 16×23 cm sa plastificiranim koricama Izlazi iz štampe početkom juna 1988 godine)
VODIČ je knjiga koja vas upućuje na druge kompjuterske knjige i časopise Osim kataloga svih kod nas izdatih naslova VODIČ sadrži i adrese izdavača i redakcija uputstava za korišćenje usluga Jugoslovenskog bibliografskog instituta opis bibliotekarskih i međubibliotečkih usluga
Specijalno za igrače kada vas neka igra muči potražite rešenja mapu ili POKE u nekom od 2600 katalogizovanih napisa!
VODIČ nabavite po pretplatnoj ceni od 13 500 dinara i to **ne izlazeći iz kuće** Nemoguće? Pozovite 011/4444-170 svakim danom (pe i četvrtkom) između 10 i 17 časova izdiktirajte svoju adresu i broj telefona platite pretplatnu cenu poštaru i očekujte knjigu Detaljnija obaveštenje o ovoj novoj i ekskluzivnoj pretplatnoj šemi dobićete preko našeg dežurnog telefona 011/4444-170
Ukoliko više volite proverene metode i šetnju do pošte uplatite 13 500 dinara poštanskom uputnicom na adresu INFOGEN (*) Poštanski fah 4, 11090 Beograd 75 i očekujte knjigu
(*) INFOGEN je nova izdavačka grupa koja deluje u saradnji i uz podršku Kluba programera Elektrotehničkog fakulteta u Beogradu Beogradske istraživačke stanice i Mladih istraživača SR Srbija

Uokvireni oglasi preko deset centimetara komercijalne cene

Razbarušeni sprajtovi



uređuje Slobodan Perović

Mirisi iz kuhinje

Upućeni ljudi komentarisali su ROCKET RANGER rekavši Indijana Džons sreće Flaša Gordona Biće da su u pravu, jer u toj igri se borite protiv nacista putujete kroz vreme Imate posla sa cepelinima i raketama u isto vreme spasavate naučnika i njegovu lepu kćer — ukratko ne znate ni gde ste ni šta vam se događa ali znate da je uzbudljivo

Zato ćete znati gde ste kad učitate YETI već vam sam naslov kazuje da ste na Himalajima Doduše nećete tražiti jetije nego svog kolegu istraživača (izgubljenog tu negde) Gomili smetala priključiće se i majka priroda (tačnije maćeha priroda), koliko da vam ne bude dosadno

S druge strane FRIGHTMARE vam nudi da sami sebi budete smetalo moraćete se probijati kroz košmare — koji su znamo, proizvod naše sopstvene podsvesti (ili lošeg varenja). Ako budete i malo srećne ruke u jednom od ukupno četiri stravična sna naići ćete i na Satanu lično

Ako vas više od košmara raduje naučna fantastika tu je CYBERNOID u kome ćete leteti po kosmosu i upucavati ste tuđina kao i hiljadu puta dosad

Pucnjavu nude i druge igre TROLL INTERCEPTOR TASK III TALADON LEE ENFIELD itd itd itd Tog zelja bilo je i biće uvek u neograničenim količinama

Zato je srećom sve manje igara koje eksploatišu temu nindža i borilačkih sportova uopšte Ako obračunamo one sasvim trećerazredne proizvode ostaje praktično samo SAMURAI WARRIOR igra puna tuče, kasapljenja i kvazijapanske muzike.

Ko zna možda je ipak najbolje povući se u miran život i učitati TOM AND JERRY

V St

DEATHRIDE

Vratolomija po vagonima zahuktalog voza Na protivnika možete da pucate ili skačete po njima (?)

Prvi nivo Na drugom vagonu čeka na vas municija Oprez kod cisterni jer bombe prašte sa svih strana a posebno se čuvajte rupe na krovu lokomotive — ako u nju upadnete loše vam se piše Dresina (džojstik pomerajte ravnomerno levo-desno) vodi u sledeći stepen

Drugi nivo Pazite da ne udarite glavom u banderu Treći nivo Iskušenje je tunel i tetok preko dva sanduka Kada dođete do vagona se vozilima obavezno skočite na gornju platformu a u ovom nivou vam je na raspolaganju i bombe (aktivira se na sledeći način, dole plus pucanje)

Četvrti nivo Na drugom vagonu pokupite zaštitnu masku (aktivira se tasterom G) inače nema prelaska preko radioaktivnog vagona

Sve u svemu, teško bez pouka ali on je već više puta objavljen

Miša Trivković

389

Usijani džojstik

*Iz Lovrana nam se javio **Božidar Alajbegović**.*

U broju 34 objavili ste uputstvo za igru Last ninja Mape koje šaljem olakšaće rešavanje prve tri nivoa

Mape objavljujemo, ali šta je sa ostala tri nivoa pogotovo sa četvrtim koji je pravi pravcati lavirint?

* * * * * *

***Dejan Sumrak** iz Beograda je zakasnio sa prikazom igre "Yogi bear" Ipak zahvaljujemo na informaciji da je za prelazak na C-64 potrebno pokupiti osam predmeta a ne samo šest kao u verziji za spektrum*

* * * * *

***Gordan Žiža** iz Maglaja pita kome bi u našoj zemlji mogao da ponudi scenarije za igre.*

Pokušaj na adresi Ing Željko Horvatek IRO "Suzy" Gružka 10, 41000 Zagreb.

* * * * * *

*Žao nam je što **Marin Perić** i **Srećko Kovačević** iz Zadra nisu stigli do kraja igre Trap door 2 Tako je prednost pripala našem saradniku **Vladimiru Jankoviću** iz Beograda*

A tu je i pouk.

* * * * * *

Dobili smo izuzetno veliki broj vaših priloga za rubriku Poukova mreža

I opet se ponovila stara boljka — mnogo toga je već objavljeno pa smo ovoga puta bili izuzetno kritični i sa tom praksom ćemo nastaviti

Dakle trudimo se da sve bude novo i neobjavljeno

* * * * * *

Znate već da rubriku "Razbarušeni sprajtovi" realizujemo zajedno sa emisijom Čip i sedam jerića koja se emituje svake subote počev od 14 02 na Prvom programu Radio Beograda.

Poznat vam je i naš zajednički telefon 011/339-070 kraj koga svakog utorka od 12 00—14.00 očekujemo da se javite

U protivnom može vam se desiti da prikaz napišete uludo. I ovoga puta je dugačka lista onih koji su ispali iz konkurencije samo zato što se sa nama nisu dogovorili a drugi su bili brži

*Pomenimo ih ipak **Đorđe** i **Mihajlo Zvicer** iz Feketića **Dario Vinš** iz Sarajeva **Branislav Tordaj** iz Zemun Polja **Srđan Stojadinović** iz Kikinde a poneki su kao **Branimir Milovanović** iz Obrenovca **Igor Stanković** iz Beograda i **Aleksandar Milićević** iz Aleksinca poslali prikaze davno rešenih igara*

Šta je tu je!

* * * * * *

***Saša Mandić** iz Vinkovaca traži nam šifru za igru Blue Max*

Koliko nam je poznato niti postoji niti mu je potrebna — dovoljno je da da gas i u pravom trenutku odlepi svog dvokrilca od piste

Što se tiče poukova koji su mu potrebni uz malo truda naći će ih u starim brojevima naših časopisa

* * * * * *

Primetili ste da smo od ovoga broja uveli rubriku Nekad bilo

Želja nam je da ovaj mali vremeplov bude putokaz novim hakerskim naraštajima.

Ali to je istovremeno i obaveza za sve one koji žele da sarađuju da vode računa o kakvim igrama pišu.

Objavljujemo samo ono što je izuzetno i što ne treba zaboraviti

U protivnom, zna se.

* * * * * *

I na kraju — svi junaci nikom ponikoše

Šta je sa igrom Jack The Nipper 2"

*Poznato nam je da je **Bojan Majer** blizu kraja a od ostalih igrača ni traga ni glasa*

* * * * * *

Poukova mreža

„Spektrum" — *Gordan Ziza*

GUNRUNNER	POKE 49171 127

„Komodor 64" — *Bojan Vujošević*

TO BE ON TOP	POKE 18124 173 POKE 11280 173
WIX-WAX	POKE 2130 173 POKE 2091 173 POKE 1920,44
BLAST PV	POKE 51216 256 POKE 60281 173
SHOT'EM WILD	POKE 4920 173
DEMOLITION	POKE 23015 44 POKE 18325 44
3D LEOUE	POKE 48213 173 POKE 44058 173 POKE 42801 173
ATV SIMULATOR	POKE 51422,256 POKE 49132,256 POKE 26132 173
PHANTOM	POKE 2132 173
DIZZY	POKE 4309 173 POKE 2589 173

„Spektrum" — *Dejan Đukić*

AIRWOLF 2	POKE 53471 D
GREAT GURIANOS	POKE 34962 183 POKE 63601,201
PARABOLA	POKE 38303 0

„Spektrum" — *Igor Pintar*

STAR WARS	pre poslednjeg USR ubaciti POKE 44625 183
FLUNKY	učitati bejzik sa MERGE "" i otkucati 20 CLEAR 24999 POKE 23797 195 RANDOMIZE USR 23760 POKE 35320 183 RANDOMIZE USR 23800

„Atari ST" *Srđan Filipović* — Svi poukovi se unose pomoću disk doktora.

JOUST	5328 004C 15 sektor 344 bajt
ARKANOID	5339 0003 0ED5 26 sektor 46 bajt
TIME BANDIT	536D 0048 27 sektor 344 bajt

EVERGREEN/NEVERGREEN

Opet dosta pisama i opet — kolaž

Evergreen Denisa Tibinca

BOULDERDASH

Mislim da ova izvanredno zanimljiva igra zaslužuje mnogo više pažnje U ulozi ste malog Rockfor da koji od nivoa do nivoa kupi određena količina dijamanata. Panekari za to treba itekako mučnuti glavom Zato se i kladim da oni koji se ne odvajaju od URIDIUMA ili 1942 ne bi prošli ni pet nivoa

Nevergreen Igora Stankovića

BAZOOKA BILL

Klisva kopija raznoraznih ubilačkih igara Grafika smešna, zvuka nema a animacija je za nulu Iz ekrana u ekran krećete se samo gornjim delom Oružje skoro na svakom koraku Baš baš vezel

Evergreen Juge Adisa

GAUNTLET II

Igra nad igrama zbog koje sam se budio sa podočnjacima nafik na šljive Jedino bi grafika mogla da bude bolja recimo 3D ali nema veze Prava stvari Kao ne PC-ju!

Nevergreen Igora Čopića i Predraga Čuluma

JACKAL

Sigurno da nema spektrumovca kome se nisu na vrh glave popele loše imitacije starog Commanda Ovo je upravo to Scenario izlizan (jedan proliv sliesije) grafika odvratna dok se zvuk može opisati samo izrazima koji nisu za štampu

Prikaz meseca

Redakcijska nagrada u iznosu od 10.000 dinara ovoga meseca pripala je **Vladimiru Pavloviću** za prikaz igre „Zig zag"

Pored njegovog u emisiji Čip i sedam jarića objavljeni su i sledeći prikazi „Inferno" **(Svetislav Jerinić)**, Out run **(Vladimir Zorić)**, Mandroid **(Vladan Đukanović)** i Trap door 2 **(Vladimir Janković)**.

RASTAN SAGA



Cilj igre je da kao kralj ratnika Rastan uništite zlog čarobnjaka Karga prerušenog u troglavog zmaja prethodno uništivši njegove mračne armije

Prvi nivo. Igra počinje u velikoj kamenitoj ravnici i sve je veoma lako jer se neprijatelji sasvim polako približavaju, ostavljajući vam dovoljno vremena da zamahnete mačem pre nego što se suviše približe. Ovo ne traje dugo jer ćete uskoro morati da se pentrate uz konopce prebacujete u Terzanovom stilu preko provalija i skačete preko pukotina. Cilj ovog nivoa je pronaći čarobni mač koji izbacuje vatrene kugle

Sve postaje mnogo teže u šumama drugog nivoa gde ćete se prvi put sresti sa nekim od stvarno opasnih neprijatelja Među njima se posebno ističe leteći čovek koji vas napada kada pokušavate da pređete reku skačući sa splava na splav Kao i na svim ostalim nivoima i ovde postoji drugi deo koji mora biti završen pre učitavanja sledećeg nivoa Smešten je u Kargovoj palati i nije mnogo teži od šume Ako pređete na treći nivo uočićete da je skoro potpuno identičan sa Kargovim zamkom pa vam neće biti težak ali ni zanimljiv Četvrti nivo Ovde ćete se zapitati da li ova igra uopšte ima kraja Neprijatelji se udružuju u jata i napadaju u talasima Pojavljuju se još veći i opasniji kao što su džinovi i himere

Drugi deo petog nivoa nudi dobrodošlu promenu u scenariju jer se zamak završava i vi ulazite u pećina Stalagmiti vam se obrušavaju na glavu i moraćete da odsečete glavu džinovskom zmaju na kraju nivoa. Ovo je teško ali nije ništa u poređenju sa višeglavim zmajem koji vas čeka na kraju Šesti nivo. Demoni i zmajevi koji bljuju vatru će dati sve od sebe da spreče susret sa Kargom-zmajem Biće vam potrebne sve borilačke sposobnosti kojima ste ovladali na prethodnim nivoima da bi ubili Karga i dobili čestitke programera

Vladimir Pavlović

DEFLEKTOR



Vaš cilj je da kroz 60 ekrana provedete laserski zrak i naravno uništavate mrake kuglice koje su vaši neprijatelji! Zrak usmjeravate tako što dođete kursorom na ogledalo koje odbija zrak držite pucanje i palicu pomjerate lijevo ili desno Ubrzo ćete primetiti mina koje su razbacane po ekranu Kada zrakom pogodite minu ili se zrak vrati ponovo u ogledalo dolazi do zagrijevanja Zagrijavanje pokazuje gornja skala u dnu ekrana a energiju donja skala Osim toga u donjem lijevom uglu se nalazi mali ekran u kome se ispisuju poruke o gubitku energije ili zagrijevanju lasera a kad zaginete ispisuje se boom boom boom U igri imate naizbježna smetala Pomalo čudno ali smetala su mrije koje kada dođu na ogledalo obično usmjeravaju zrak u mine. Njih možete uništiti tako što dođete kursorom na njih i pritisnete pucanja

Uništena smetala se poslije izvjesnog vremena ponovo pojava Tu su i razna pomagala kocke pokretna ogledala i automatske skretnice Kada na kocke dođe zrak razbacuju ga u raznim pravcima Pokretna ogledala se vrte u krug i kada zrak dođe na njih pohiju ga oko sebe Automatske skretnice prebacuju zrak sa jednog na drugo mjesto I morate paziti kako zrak usmeravate u tu skretnicu Na svakom novom ekranu obnavljaju se vaša tri života i dobijate bonus za svaki život koji vam je ostao iz prethodnog ekrana Bonus dobijate i za energiju koja vam ostane Kada uništite sve kuglice na ekranu cigla ili mina ispred izlaza će nestati i moći ćete da izađete. Morate paziti da zrak pada u izlaz pod pravim uglom Igra ima klasičan meni sa hi-score tabelom a i ne obiluje baš zvukom Nadam se da će vam se dopasti Tu je naravno i pouk za besmrtnost POKE 34473,61 sa ona kojima ne ide od ruke Svi ekrani imaju svoje šeme koje su sve iste pa bi bilo dobro da ih pamtite.

GOTOVO JE, GOTOVO



GARFIELD IN BIG FAT HAIRY DEAL

Nakon serije crtanih filmova debelog mačka Garfilda pojevila se i istoimena igra Garfildovu macu (tj djevojku) uhvatio je šinter i zatočio je u gradskoj živodernici a Garfield namjerava da je spasi

U toku igre nailazite na veoma korisne predmete koje možete podići džojstikom nadole Možete izabrati da li ćete hodati na dvije ili četiri noge Postoji i vrlo elegantno urađeni pokret šut Postižete ga pucanjem +džojstik u pravcu kretanja Pomoću njega možete otvoriti razne predmete (kovčeg senduk) pa čak i da se otarasite dosadnog psa Odiea Njegovu pažnju možete odvući i pomoću koske Žetke ili ući u druge sobe ili pak proći kroz rupe ograđa postavite se ispred prolaza i pritisnite gore+pucanje Možete proći kroz otvorena vrata a kroz zatvorena samo ukoliko posjedujete ključ Njega ćete naći u podrumu kuće u koju ulazite kroz rupe na podu Sakriven je u kovčegu koji prethodno morate otvoriti šutem Učinite to brzo jer ako ga dohvati ogromni pacov koji se ovuda mota gotovo je Osim podruma tu su nadovezane gradske kanalizacije Neke su toliko mračne da ih možete proći samo lampom

U toku igre važno je da obratite pažnju na osnovne podatke u dnu ekrana. To su sitost predmet koji nosite i snaga Sitost nadopunjavate hranom (lazanje kobasice) a snagu ili umor nadopunjavate mirovanjem Snagu gubite krećući se a i Odie vam je oduzima Ukoliko nećete da čekate dok vam se nadopuni snaga jednostavno se popnite na Johnov sto (koji doručkuje) i srknite kavu iz njegove šalice Možete to učiniti samo dok John gleda na stranu Inače ako vas vidi Garfild će se uplašiti i pobjeći sa stola U toku igre pojavljuju se i prodavači u raznim dućanima grada (mesar) Kod mesara možete nabaviti kobasice kod novinara novine itd Neki će vam narediti da odmah izađete van a Garfild će ih automatski poslušati Za igru je potreban bistar um da biste je riješili do kraja. Sretno!

Stjepan Zeilder

STAR WARS

Konačno smo dočekali da i „Zvezdani ratovi" dođu u čipove naših mikrića Zadatak je uništiti Imperijalnu svemirsku borbenu stanicu znanu pod imenom „zvezda smrti"

Igra se odvija u tri nivoa

1 — put do zvezde smrti
2 — borba na njenoj površini
3 — borba u njenoj unutrašnjosti

Ekran nije prenatrpan Tu su poeni broj štitova stepen težine igre i sl Pomeranjem kursora nišanite na dušmane i pomerate brod (kao kod simulacija)

Smetala

Ima ih tri vrste. Imperijalni lovci se pojavljuju na prvom nivou Dovoljan je jedan pogodak Vatrene kugle ispaljuju kako lovci tako i laserski tornjevi Morate ih uništiti inače ode jedan od devet života Razne prepreke u vidu laserskih tornjeva i bunkera se pojavljuju na drugom i trećem nivou i neuništivi su

I evo jednog malog trika (doduše na prvom stepenu težine) Treći nivo prolazite bez problema ako kursor dovedete u donji desni ili levi kut Potom ćete imati zadovoljstvo da vidite kako „Zvezda smrti" ide do vraga

Tihomir Dmitrović

390

MORPHICLE

Iako se sastoji od samo 3 nivoa igre nije nimalo laka

PRVI NIVO

Upravljate vozilom koje ima mogućnost da se pretvori u avion. Vozilo se kreće krivudavim drumom punim raznih prepreka. U ograničenom vremenu treba da „doturate" vozilo do stanice. Pritiskom na pucanj vozilo se pretvara u letjelicu i preleće prepreke. Uz malo vježbe ovaj nivo vam neće predstavljati veće poteškoće.

DRUGI NIVO

Ovo je daleko najteži nivo. Nema drugih protivnika osim vremenskog ograničenja. Upravljate svemirskim brodom koji treba da se izvuče iz vrlo zamršenog lavirinta. Na desnom donjem uglu ekrana se nalaze strelice. Strelicu morate pomaknuti u onaj smijer kuda želite ići se brodom.

Na lijevoj strani ekrana je prikazana mapa lavirinta a dio koji ste prešli se oboji žutom bojom.

Uložiti ćete puno truda za prelazak ovog nivoa ali se nećete pokajati kad vidite slijedeći nivo.

TREĆI NIVO

Daleko najbolji i najzanimljiviji nivo.

Zadatak vam je da složite slagalicu (PUZZLE) u što kraćem vremenskom roku. Njenim slaganjem deaktivirate bombu i po tko zna koji put spasavate svijet od uništenja.

Neka ostane tajna ono šta prikazuje slagalica. Potrudite se i otkrijte to sami.

Prijatna zabava!

Alsjbegović Božidar

Poukova mreža

„Komodor C—16"

Saša Popović

Poukovi se unose u memoriju odmah po učitavanju programa i ulaska u monitor istovremenim pritiskom na tastere RUN/STOP i RESET.

GUZZEER	otkucajte D1798 i <RETURN> i pojaviće se. D1798 A9 01 LDA #$01 zadnja dva bajta prepraviti na #$00 D1C62 zadnja dva bajta prepraviti na #$00 SYS 9856
PUNCHY	D1921 zadnja dva bajta prepraviti na #$00 DA2FF zadnja dva bajta prepraviti na #$99 SYS 4112
KIKSTART	D21AB zadnja dva bajta prepraviti na #$00 SYS 8192
BIOOSVAJAČ	D15FB zadnja dva bajta prepraviti na #$99 SYS 5600
DEATH RIVER	D1135 prepraviti zadnja dva bajta na #$99 SYS 16351
DIRTY DEN	D178F prepraviti zadnja dva bajta na #$00 SYS 4112

Iz monitora se izlazi sa X <RETURN> pre startovanja programa.

„Amstrad"

Domagoj Marić

TRANTOR — vreme i neranjivost

```
POKE &1CA &C9: CALL &180
POKE &2A99 0: POKE &0C6A 0
POKE &0C27,0: CALL &853E
```

FREDDY HARDEST 1

```
10 OPENOUT „D" MEMORY &1918
20 LOAD "
30 POKE &8BEA 0: BESMRTNOST
40 CALL & 181A
```

FREDDY HARDEST 2

```
10 OPENOUT "D" MEMORY &856
20 LOAD "
30 POKE &6APC &B7 BESMRTNOST
40 CALL &857
```

BATTY

```
10 OPENOUT "D" MEMORY &856
20 LOAD "
30 POKE &6867 0: BESMRTNOST (1 IGRAČ)
40 POKE &6975,0: BESMRTNOST (2 IGRAČ)
50 CALL &4802
```

TERRA COGNITA

```
10 OPENOUT "D" MEMORY &FFF
20 LOAD " &1000
30 POKE &2130 0: BESMRTNOST
40 FOR X=&A000 TO &A00D
50 READ A
60 POKE X,A
70 NEXT
80 CALL &A000
90 DATA &21 &0. &10. &11 &80 &1 &1 &10 &3B &ED, &B0 &C3 &80, &1
```

„Spektrum"

Nebojša Lazović

Svi poukovi se ubacuju ispred RANDOMIZE USR ili PRINT USR

ATHENA	POKE 48853 0: POKE 58861,0: POKE 51561 0: POKE 52593 0: POKE 55594,0
THE PLOT	POKE 44658 183
3DC	POKE 35189 0 POKE 34635 0: POKE 35438 0
THE TUBE	POKE 65368 245
MERCENARY	POKE 33356 0 POKE 39724 0
DRILLER	POKE 47904 0. POKE 49021 0 POKE 49022 0: POKE 49424 0: POKE 49425 0
NEBULUS	POKE 32913 0
RYGAR	POKE 60709 0: POKE 61577 0
BOSCONIAN	POKE 33356 0 POKE 39724,0

ZIG-ZAG

Vi kontrolišete mali svemirski brod a zadatak je da sakupite 8 fragmenata magičnog kristala razasutih u 16 zona i zatim bekstva što se postiže odlaskom u tzv. End zone.

Igra ima dve verzije — arkadnu avanturu i čisto arkadnu igru — što znači da možete da birate da li ćete biti zakleti istrebljivač vanzemaljskih neprijatelja ili neustrašivi istraživač. U obe verzije ćete naići na mnoštvo različitih hodnika raskrsnica i prolaza kojima je najbolje prilaziti polako i oprezno jer sudari sa zidovima nisu preporučljivi a u arkadnoj verziji se uz to nikad ne zna šta može naići u susret. Na svim raskrsnicama i okukama ćete nailaziti na prizme kroz koje morate proleteti ako želite da skrenete, jer vaš brod bez njih ne može napraviti okret od 90 a kamoli 180 stepeni.

U arkadnoj avanturi nema neprijatelja, ali zato ima mnoštvo zagonetki u vidu skrivenih skretanja i prizmi koje postaju vidljive i samim tim i upotrebljive tek kada svojim brodom pređete preko senzorskih ploča postavljenih u podu. Ako aktivirate pogrešnu ploču mogu da vam se dese neke vrlo neprijatne stvari ali i tu morate da učite na svojim greškama jer se takve ploče nasumično pojavljuju. Još treba da se čuvate magneta postavljenih u zidovima ili na plafonu koji vam privlače brod kao i tzv. „zona smrti" koje vam vrlo brzo oduzimaju energiju. Obe ove prepreke postoje u svim verzijama igre.

Sve raskrsnice su otvorene sve prizme vidljive. Nema logičkih problema već samo uništavanje neprijatelja od početka do kraja. Postoji sedam tipova neprijatelja — svaki sa po tri različite napadne tehnike. Svaki put kada uništite nekog neprijatelja dobijate određen broj kredita što se vidi u statusnom delu ekrana. Kada ih sakupite dovoljno moći ćete da stanete i u najbližoj prodavnici opreme kupite neki dodatak za svoj brod.

Dodaci koje možete kupiti su:

— **Instant U-turn**, koji vam omogućava da bilo gde napravite okret od 180 stepeni. Košta 5 kredita i možete ih kupiti do 99 odjednom.

— **Zapper**, koji uništava sve neprijatelje u području koje se proteže i izvan ekrana. Košta 10 kredita maksimalno 99 odjednom.

— **Mapper**, koji služi za dobijanje mape zona sa obeleženim pređenim putem i lokacijom broda. Košta 20 kredita maksimalno 99 odjednom.

— **Fast Missiles**, imaju manji domet od običnih raketa ali veću brzinu i razorno dejstvo. Koštaju 40 kredita, samo jedno pakovanje odjednom.

— **Riser**, koji prisiljava sve neprijatelje da se zalepe za plafon gde vam ne mogu nauditi osim ako vas ne privuče magnet. Košta 80 kredita.

— **Shield**, energetski štit traje 5 minuta. Štiti vas od svih neprijatelja osim od „zone smrti". Košta 50 kredita.

Kupljeni dodatak upotrebljavajte tako što ga prvo namestite na osvetljeni deo menija (lista se bilo kojim tasterom sa desne trećine statature) i pritisnite dugme za paljbu. Neki od dodataka mogu se upotrebiti samo jednom i onda nestaju, neki traju određen broj minuta a neki kao infracrveni i rendgenski vizir koji se mogu kupiti samo u specijalnim prodavnicama traju dok ne izgubite život.

Vladimir Pavlović

Sokoban

Sve što vam je potrebno da biste uspešno igrali ovu igru je čvrsta volja i napregnute vijuge. Nalazite se u ulozi radnika koji ima zadatak da sve sanduke koji su na ekranu smesti na svoja mesta koja su obeležena kružićima. Možda vam to izgleda lako ali da biste ostvarili cilj moraćete dobro razmišljati. Samo jedan pogrešan korak odnosno samo jedan pogrešno pomeren sanduk može vam onemogućiti da skupite sve sanduke i tako obavite svoj posao. Odjednom možete gurati samo po jedan sanduk. Pritom pazite da ne gurnete sanduk uz zid jer ga više nećete moći odvojiti od zida.

U dnu ekrana se nalaze tri brojača koji broje vreme od početka igre koliko ste pokreta napravili i koliko puta ste pomerali sanduke. U verziji za „komodor 64" tipka F3 vraća igru na početak a tipka F1 blokira računar pa ako to ne želite nemojte je pritiskati.

Ideja igre je dobra ali autor se mogao potruditi da je bolje razlizuje jer grafika je veoma loša a zvuka skoro da i nema.

Ćazim Dervišević

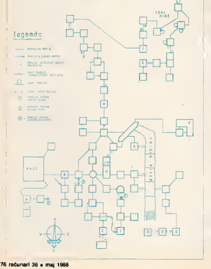
ZORK I THE GREAT UNDERGROUND EMPIRE

Ova aventura se pojavila u nas sa velikim zakašnjenjem, evo nekoliko uputstava za lakše igranje i rješavanje igre

— **jaje**
Samo ga lopov može otvoriti, pa mu ga stoga daj što prije
Sačekaj malo i onda mu ga uzmi
— **trolovi** (soba označena na mapi slovom B)
Ukoliko imate poteškoća, upotrijebite mač
— **loud room** (soba označena slovom H)
Otipkaj riječ eho (TYPE ECHO)
— **Entrance to Hades** (soba C)
Pozvoni na zvono (RING BELL), upali svijećnjake koje ćete naći u sobi I (LIGHT CANDLES) te pročitaj crnu knjigu
(READ BLACK BOOK)
— **Drafty room** (soba E)
Bolje ćeš vidjeti ako upališ lampu koju ćeš naći u košari u sobi J. Ukoliko je okrećeš (TURN LAMP) svjetiosi možeš pojačivati ili smanjivati
— **Stjenoviti prolaz (TEH STONE BARROW)**
Kroz njega možeš proći samo ako imaš više od 350 poena
— **DAM BASE (soba G)**
Ovdje se nalazi hrpa plastike (PILE OF PLASTIC) koju možeš da naduvaš ako upotrijebiš zračnu pumpu (AIR PUMP). Nju ćeš naći u sobi H
— **Maintenance room (soba F)**
Odlika ove sobe leži u tome što se tu nalazi neprsnuta cijev
Ukoliko pritisnete plavo dugme (THE BLUE BUTTON), kapanje će se pojačati. Pretpostavljem da je tu potreban neki predmet da zaustavi kapanje
— **Kiklopi (soba A)**
Oni se smrtno plaše imena svojih velikih neprijatelja Odisdje odnosno Ulisa. Njih ćeš dobiti proučavajući naredbe u crnoj knjizi
— **Duga (soba D)**
Kada se nađeš na njenom kraju, probaj mahati skiptarom (WAVE SCEPTRE).

Riječ dvije i o mapi. Važne sobe su označene slovima radi lakšeg snalaženja. Prolazi gore dolje su označeni slovima U i D, a prostorije koje nisam mogao da nacrtam, označio sam brojevima. Do kraja još nisam došao ali sam blizu

Jurić Siniša



MANDROID

Radnja igre se dešava u dalekoj budućnosti na jednoj od mnogobrojnih planeta u galaksiji. Mandroid je pola-čovjek, pola robot, kreatura dušu dale za poslove koji se graniče sa zakonom. On dolazi na planetu na kojoj caruje kriminal sa ciljem da je uništi po svaku cijenu. Dolazi do povjerljivih dokumenata koji sadrže nacrte odbrane planete i njegov zadatak je naizgled gotov. Ali, kao što rekoh, planetom vladaju kriminalci. Tako se desilo da je lopov po imenu Kris Ouker „pozajmio" Mandroidove nacrte. Tu se priča završava i počinje igra. Igra nije obična pucačina, već je potrebno i malo logičkog razmišljanja. Kupovina i prodaja od velikog su značaja za uspjeh vaše misije. Možete ljubazno razgovarati sa likovima koje susrećete, a ako ne ide milom onda će ići silom. Uglavnom, igre kakve odavno nije bilo

Možete koristiti više vrsta oružja, a na početku ste opremljeni samo pištoljem (hand gun). Do ostalog naoružanja dolazite borbom ili trgovinom. Pritisak na taster F1 daje vam meni za izbor oružja. Palicom odaberite ono koje želite i pritisnite FIRE. Veoma korisno oružje predstavlja **REMOTE**, robot koji vam služi za izviđanje. Možete ga poslati u susjednu sobu da ispita teren, a takođe možete aktivirati bombu koju on nosi. Ova bomba uništava sve živolne oblike izuzev vas. Pritiskom na taster F3 dolazite u meni sa sledećim opcijama.

GREET — pozdrav. Ovom opcijom pozdravljate likove sa kojima želite da razgovarate. Ako nisu raspoloženi za razgovor, pojaviće se poruka NO ANSWER

QUESTION — pitanje. Ovim saznajete informacije od osobe sa kojom razgovarate. To mogu biti podaci o tome gdje da idete, podaci o lopovu koji vam je ukrao nacrte i tako dalje

BRIBE — podmićivanje. Neke životne forme na ovoj planeti nijesu baš mnogo obradovane vašim prisustvom i odbijaju uspostavljanje kontakta. Ali tu vi, prema onoj narodnoj „Para vrti gdje burgija neće", postajete velikodušni i dajete 50 kredita za informacije. Pazite se možda uludo bacate pare

THREATEN — prijetnja. Ako neće milom, hoće silom. Poslužite se ovom opcijom kada ništa drugo ne pomaže.

BUY — kupovina oružja i opreme
SELL — prodaja istoga

Željenu opciju opet birate palicom za igru i tasterom FIRE. Pored opcija, prikazane je i mapa planete koju dobijate od likova koje srećete, vaša energija, kao i novčana sredstva koja vam stoje na raspolaganju. U početku je to 250 kredita, ali imate otvoren žiro-račun kod Galaktičke banke sa kojeg možete podići novac. Do njega možete doći i prodajom dijela oružja ili opreme, kao i kockanjem. Naime, u nekim prostorijama se nalaze pravi automati za kockanje JACK POT, kod nas poznat kao „Jednoruki Džek". Krećete se po površini planete na kojoj se nalaze zgrade. U njih možete ući ukoliko postoje vrata. U zgradama se nalaze automati za kocku, a na površini planete su bankovni terminali pomoću kojih podižete ili ostavljate novac. U kretanju površinom planeta možete se poslužiti vozilom. Kada uđete u njega, u dnu ekrana se pojavljuju skale koje prikazuju brzinu, gorivo i stepen oštećenja automobila. Kada mirujete ova poslednja skala opada. Takođe kada nijeste u vozilu prilikom mirovanja vaša energija se obnavlja.

Najvažniji likovi koje susrećete u igri su

MANDROID. To ste vi. Ukoliko se vaša misija oduži stvaraju se novi mandroidi koji će vas napasti. Veoma su opasni.

GUARD. Straža. Patroliraju po kompleksu zgrada, naoružani su i ne mogu se potkupiti, niti se mogu saznati bilo kakve informacije od njih. Poznaćete ih po rančo na leđima

THIEF. Lopov. Ako je sam, ponaša se prijateljski i možete kontaktirati sa njim. Ako mu stanete na žulj, 'em će vam ukrasti pare, 'em će pokušati da vas ubije. Sličan je stražarima, ali ima manji ranac na leđima i traku oko glave

PIRATE. Svemirski gusar, naoružan, ali se dosta lako uništava. Možete trgovati sa njim i doznati informacije, ako to želite. Ima na leđima veći ranac nego stražari

BANDITE. Odmetnik. Nema ranca, već samo traku oko glave. Može se kontaktirati sa njim i nije mnogo opasan.

ARMOURER. Oružar Ako imate dovoljno para od njega možete da kupite oružje Uvijek se nalazi u zgradama i za razliku od drugih likova na kraće se već stoji skrštenih ruku Preporučujem vam da od njega nabavite REMOTE LASER CARABINE i MACHINE GUN Ova oružja su najdjelotvornija

DROID Androidi koji će vas uvijek napasti ali sa lako daju uništiti

ROBOT Sličan androidu ali opasniji

WOMEN Prostitutke koje stoje uz zidove zgrada i nude svoje usluge Od njih možete saznati dosta Nijesu opasna

CHRIS AUKER Lopov koji vam je ukrao dokumenta Naoružan je i opasan Lako ćete ga poznati jer je obučen u potpuno crno odijelo

MAX Vladar planete koga morate da smaknete ukoliko želite da uspješno dokrajčite vašu misiju On je potencijalni kupac ukradenih nacrta

Evo kako da završite igru Lutajte planetom sve dok na sretnete Krisa Oukera Nemojte ga uništiti već ga pratite Dovešće vas do Maksa Zatim sredite obojicu pokupite dokumente i vaša misija je gotova

Jednostavno zar ne?

Vladan Đukanović

BOCCE BALLS

U igri postoji najmanja kugla (bulin) i po dvije veće kugle (boće) za svakog igrača Jedan od igrača baca bulin a zatim i svoju boću što bliže bulinu Drugi igrač nastoji baciti svoju boću još bliže ili pokušava udaljiti protivničku Onaj igrač čija je boća dalje od bulina pokušava ponovo Ako ne uspije protivnik ima šansu da osvoji dva boda s tim da ima dvije boće najbliže bulinu Slijedeću partiju pobjednik prvi počinje Igru može igrati do 10 20 ili 30 bodova što možete izabrati u meniju

Poslije startanja programa pojavljuje se meni za izbor terena

1 INDOOR — Igra u dvorani teren je ravan i čist

2 ON FIELD — Igra na otvorenom kamenje i neravan teren

3 PRACTICE — Uvježbavanje (objasnit ćemo ga nešto kasnije)

Možete igrati protiv kompjutera ili protiv prijatelja Ukoliko igrate protiv kompjutera možete izabrati jedan od tri stupnja težine (NORMAL AMATEUR EXPERT) Nakon svih priprema napokon počinjemo

Sasvim lijevo je izgled terena na kojem vidite položaj igrača (ispod crte) te kursor koji određuje smjer bacanja U sredini-dolje je slika igrača (nema važnosti za igru) Desno od slika su imena igrača i bodovi Prozor označen sa POWER određuje snagu bacanja a SPIN rotaciju boće Iznad slike sa igračem je glavni prozor sa opcijama Birate ih pomicanjem džojstika gore-dolje

DIRECTION — pomiče kursor za smjer bacanja

SPIN — određuje rotaciju boće Jačina rotacije je prikazana u prozorčiću SPIN

POSITION — mijenja poziciju bacača boće tj vas Kada ste napokon sve namjestili pritisnite pucanje i držite ga toliko dok se skala u prozoru POWER ne popne do željene visine

Vratimo se sada na opciju PRACTICE Ona služi za uvježbavanje raznih pozicija Na početku na raspolaganju su vam tri opcije

SELECT — birate boće

SET — pomicanje prethodno izabrane boće

DONE — prelazite u standardni meni za bacanje Ovom opcijom možete boće razmjestiti po terenu i vježbati bacanje Nakon bacanja pojavljuju se nove tri opcije

NEW SCHEME — ponovo prelazi u PRACTICE mod

SAME AGAIN — ponovo bacanje sa istim rasporedom boće

OPTIONS — prekida PRACTICE i vraća nas na početnu sliku

Uprkos skromnom zvuku igra će vas zaokupiti radi originalne i prihvatljive ideje

Rebić Ivan i Grbavac Zoran

GOTOVO JE, GOTOVO

TERRAMEX



Ogromnom brzinom ka zemlji leti ogromni meteor ali sve nije izgubljeno Tu je Albert Ajnštajn koji velikom polugom može da odbije meteor od zemlje ali u svojoj laboratoriji nema potrebne delove za aktiviranje instrumenata koji pokreću polugu E te delove ćeš mu ti doneti Na izboru imaš pet likova ali samo jednog od njih možeš da kontrolišeš a njega biraš na početku igre Do udara meteora u zemlju ostalo je još trideset dana Pa da počnemo

Nalaziš se u prirodi Idi desno i uzmi usisivač On je koristan samo ovde pa se zato vrati malo levo i upotrebi usisivač da bi poleteo nagore Leti na levo pa uzmi bure do koga dolaziš skokom Odleti na desno pa uzmi predmet koji liči na kavez slona šipke kišobran i bicikl sa jednim točkom Vrati se na startnu lokaciju pa kreni desno Uzmi frulu I sada možeš da prelaziš preko zmija Skoči u bunar i dođi do provalije Za prelaz preko provalije koristi predmet u obliku kaveza Napraviće se most preko koga možeš da pređeš. Posle ovoga ponovo dolaze zmije pa ponovo uzmi frulu Samo pazi ovde nemoj da silaziš dole jer će te zmije ščepati već pređi preko gornje prečage

Kada to pređeš doći ćeš do odskočne daske Skoči na nju i odbaciće te nagore Tu uzmi meh i štapove za polo Vrati se do ulaza u bunar pa kreni levo i uzmi foto aparat i još jedan predmet Izađi iz bunara Idi skroz desno i naći ćeš na balon. Iskoristi meh da napumpaš balon i polatećeš gore Odmah kreni desno Izađi iz balona (pritisni S) i padni dole Jako što ćeš kišobranom usporiti pad Doći ćeš do topa Prazno bure koje stoji pored njega je zamka i iskoristi ono koje si pre toga pokupio. Top će te odbaciti preko provalije a ti uzmi kišobran pa padni dole Idi na levo E ovde se avantura razgranava tj. ovde svaki lik radi različitu stvar

Predmet koji ovde treba da pokupiš je Ajnštajnova formula koja je neophodna za dalji nastavak avanture

1 Ako si igru počeo kao Kinez put ka formuli će ti čuvati zeleno čudovište Uzmi aparat i dođi pored njega Kada blic sevne ono će pokriti oči a ti ćeš moći da uzmeš formulu

2 Ako si počeo kao traper put će ti preprečiti drveni konj Uzmi štap za polo i uzjaši konja Put ka formuli je slobodan

3 U ulozi istraživača uzmi krug koji je bio u blizini frule Odmakni se malo od prečage koja stoji na kraju ekrana i istraživač će baciti krug na prečagu i slomiti je Sa neba će pasti formula

4 Kao Nemac put ka formuli će ti preprečiti Policajac Po starom dobrom nemačkom običaju uzmi kriglu piva (koje se nalazi na tom spratu desno) i ppol zajedno sa policajcem Prolaz ka formuli je slobodan

5 U ulozi Engleza na putu će ti se naći provalija preko koje je razapet konopac Sada izigravaj akrobatu pa uzmi onaj bicikl sa jednim točkom i pređi preko konopca Formula je tvoja

Sada je do kraja rešenje isto za sve likove Uzmi formulu pa idi levo u laboratoriju Stani na pokretnu platformu pa se odvezi gore Idi DESNO i upašćeš u antigravitacionu sobu Ovde kako znaš i umeš uzmi predmet koji stoji na sredini sobe pa se vrati istim putem do ulaza u laboratoriju Idi još dole sa platformom a onda uđi u levi prolaz Pazi ovde se nalazi viseći most a preko njega se prelazi neprekidnim skakanjem jer ako budeš puno hodao tvoj čovečuljak će dobiti vrtoglavicu Uzmi predmet pa skoči na odskočnu dasku i ona će te odbaciti na gore Tu uzmi polugu idi na levo pa sa kišobranom padni dole Nabasaćeš na teleport Njega aktiviraš tako što ćeš na levu platformu staviti predmet koji si uzeo u antigravitacionoj sobi Uđi u teleport Kada se stvoriš u drugom teleportu idi odmah levo i uzmi kazan Vrati se ali nemoj još da ulaziš u teleport već idi desno Stavi polugu u krug koji se nalazi na zidu Uđi u korpu i prevezi se preko provalije Tamo uzmi narandžastu loptu pa se vrati istim putem ali pre nego što uđeš u korpu uključi prekidač Sada uđi u teleport i naći ćeš se u laboratoriji Idi levo pa dva ekrana dole pa desno dok nedođeš do zadnjeg predmeta koji ti treba Pokupi ga Idi levo i popni se merdevinama Dođi do nakovnja i od šipki napravi krst Popni se gore i krstom sprži Drakulu Da bi prešao preko električnog polja upotrebi Ajnštajnovu formulu Posle ovoga naći ćeš se licem u lice sa naučnikom Proveravajući instrumente on će ti tražiti jedan po jedan predmet Nemoj slučajno da mu daš neki drugi predmet (koji nije tražio) jer će ga to razljutiti Predmete mu daj tačno po redu i kada mu budeš dao i poslednji predmet kompletiraće instrumente koji do sada nisu radili a ti uživaj u animiranoj sceni koja je pripremljena za kraj

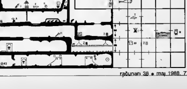
Veoma važno je da znaš sledeće Predmet koristiš pritiskom na S Predmete nameštaš sa 1 i 2 Ni slučajno nemoj da koristiš slona dok se voziš balonom jer će se desiti nešto nepredviđeno Za lakše igranje koristi mapu

Đorđe Perović i Vladimir Janković

LEGENDA ZA MAPU

B — BURE
S — SLON
Š — ŠIPKE
K1 — KIŠOBRAN
K — KRUG
F — FRULA
U — USISIVAČ
A — FOTO APARAT
P1 — POLUGA
K2 — KAZAN
N — NAUČNIK
N1 — NAKOVANJ
M — MEH
P — ŠTAP ZA POLO
L — TELESKOP
T — KLJUČ ZA TELEPORT
D — DRAKULA
K — KAVEZ
Z — PREDMET KOJI TREBA NAUČNIKU

TERAMEX



GARY LINKER'S SUPERSTAR SOCCER

Kad se igra učita susrećemo se s menijem koji sadrži jedanaest opcija.

LEAGUE STATUS tom opcijom dobivate podatke o količini novca kojeg možete potrošiti za razmjenu igrača koliko se utakmica igra po sezoni koja vam je to sezona i koliko ste utakmica odigrali u prvenstvu i kupu

TEAM HISTORY ova opcija je statistika pobjeda poraza neriješenih utakmica datih i primljenih golova osvojenih bodova itd Pritiskom na tipke 6 i 7 možete vidjeti i statistiku ostalih klubova

GAME SCORES služi za pregledavanje rezultata ligaških i kup utakmica svih liga (1—4)

DIVISIONS služi za pregledavanje tablica svih liga (1—4)

LOAD/SAVE mislim da ovu opciju nije potrebno posebno objašnjavati

RECRUIT PLAYER ovom opcijom igraču mijenjamo poziciju u timu npr lijevo u desno krilo a također možemo izvršiti i zamjenu igrača

TRADE PLAYER ima tri opcije — return (povratak na glavni meni) try trade (opcija za zamjenu igrača) i view teams (služi za pregledavanje igrače u drugim timovima) Kod igrače je vrlo bitna njegova vještina (skill) a manje starost (age) Zamjena se vrši ovako. u drugim timovima nađemo igrača koji nas zanima. Zatim izaberemo opciju try trade i kompjuter nas obavještava da je početni ulog x' novčanih jedinica na što možete pristati ili odustati ako vam je previše Zatim izaberete igrača iz svog tima kojeg želite zamijeniti za igrača iz drugog tima kojeg ste odabrali za zamjenu Zatim vas kompjuter pita koliko ćete još doplatiti Na to možete odgovoriti s Ø ili doplaćujete koliko želite Zatim se klubovi dogovaraju i kompjuter vas obavještava da li je zamjena prihvaćena ili nije

IMPROVE TEAM opcija služi za treniranje tima Na pitanje koliko novaca želite utrošiti na trening možete odgovoriti s Ø ili odrediti količinu novca za trening

SET UP TEAM opcija služi za sastavljanje tima igrači su označeni trima bojama. crvenom žutom i bijelom Crvena boja je znak premorenosti Igrače mijenjate tako da kursor dovedete na igrača kojeg želite zamijeniti i pritisnete Ø Zatim ćete dobiti dvije opcije MOVE i SWAP Opcijom MOVE izabranog igrača mijenjate s jednim od dvojice igrača s popisa označenih zeleno a opcijom SWAP s jednim od dvojice igrača označenih sa SUB1 i SUB2 ili sa drugim igračem koji je već u timu. Iz opcije se izlazi sa QUIT

NEW LEAGUE na pitanje da li će svi rezultati biti izbrisani odgovarate sa OK ako želite krenuti ispočetka a sa CANCEL da se vratite na glavni meni Ako ste odgovorili sa OK možete izabrati teritorij na kojem će se igrati utakmica (Engleska SAD) i broj utakmica koje će se odigrati u jednoj sezoni Zatim se opcijom FINISHED vraćate na glavni meni

NEXT GAME sastoji se od manjeg menija — sa GAME SPEED mijenjate brzinu igre Sa CENTRE birate da li ćete vašu momčad voditi vi ili kompjuter Opcijom COACH određujete kako ćete se braniti i napadati ili će i to obavljati kompjuter Sa PLAY GAME počinje utakmica Umjesto PLAY GAME ponekad će vam se pokazati opcije SET UP TEAM ako vam je koji igrač premoren. Izaberete li opciju PLAY GAME na pitanje da li ćete promatrati meč možete odgovoriti sa da ili ne Sa CANCEL se vraćate na glavni meni

I na kraju jedan savjet: pustite da kompjuter obavlja sve poslove oko utakmica jer ako budete pokušali igrati ili promatrati utakmicu uvjerit ćete se da traje doslovno 90 minuta Tipka za prekid igre ne postoji

U cjelini igra je odlična i predstavlja pravo osvježenje među brojnim pucačkim igrama koje zahtijevaju samo brže reflekse. a posjeduje još jedno bitno svojstvo — adiktivnost.

Tomislav Klemenčić

GOTOVO JE, GOTOVO

RAMPAGE



Glavni junaci ove zanimljive igre su tri velika čudovišta — GEORGE LIZZY i RALPH U potrazi za hranom zabavljaju se rušeći zgrade i druge objekte u gradovima dok stanovnici nastoje da ih unište za sva vremena Ali zato ste vi tu da boravak ovih čudovišta u gradu napravite što bezbednijim Možete da upravljate sva tri čudovišta no ja bih vam preporučio da preuzmete ulogu samo jednog (recimo GEORGE) a uloge LIZZY i RALPH a prepustite kompjuteru.

Zgrade rušite na jednostavan način Penjući se bilo sa leve ili desne strane pravite pukotine pomoću vaših snažnih pesnica Pri tome možete da koristite dva različita udarca Prvi udarac (taster levo desno+pucanje) vrlo je efikasan i pomaže vam naročito pri skupljanju hrane Drugi udarac (taster dole+pucanje) koristi vam da što brže razrušite zgradu Kada napravite određeni broj pukotina sa jedne i druge strane zgrada počinje da puca i da se ruši Nastojte da je onda napustite! Čudovišta raspolažu sa određenom količinom životne energije koja nije naročito velika Zato je potrebno da vaše čudovište stalno snabdevate hranom

Skupljajte je dok pesnicama pravite pukotine u zgradi Izbegavajte televizore otrove novac — oni loše utiču na životnu energiju Preporučujem vam šolju mleka komad mesa pečeno pile bocu i čašu U vaš jelovnik možete da unesete i negostoljubive stanovnike grada ali oni ne utiču na vašu životnu energiju Neprijateljske oružane snage čine borbeni helikopteri (čuvajte se njihovih rafala!) avioni tenkovi i patrolna kola Najviše briga zadaće vam stanovnici Oni neprekidno sa prozora bacaju razne štetne predmete Zato je najsigurnije da na početku prvo njih uništite a zatim pristupite rušenju Još jedan koristan savet zgrade rušite počevši od donjih spratova Kada pređete svih deset gradova počinjete sve iznova. Na kraju — POKE za besmrtnost!!!

Poukova mreža

Spektrum

Ivan Mirčevski

SECTOR 90

bezbroj života i vreme

```
15 MERGE
16 POKE 23784 195
30 POKE 36584 0: POKE 36585 0: POKE 36586 0
40 POKE 36420,182 RANDOMIZE USR 23767
```

SALAMANDER

bezbroj života

```
MERGE    a zatim POKE 23819,195
10 POKE 48260 0 POKE 48261 0: POKE 48262 0
20 RANDOMIZE USR 23822
```

PHANTIS

Novo remek-djelo španskih programera Ovoga puta si djevojka koja spašava svog momka zarobljenog na tuđinskoj planeti U meniju stisni T za redefiniranje tipki J za joystick E za start Igra se sastoji iz dva dijela U prvom ti se probijaš do unutrašnjosti planete

1 nivo — u svemiru Samo pucaj i izbjegavaj neprijatelje Meteore treba pogoditi dvaput

2 nivo — iznad planete Najteži nivo Neprijatelji napadaju iz svih praveca Velike kugle koje izlijeću iz vulkana ne možeš upucati pa ih bolje izbjegavaj

3 nivo — u podzemnim hodnicima Veoma lak

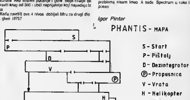
4 nivo — jašeš kroz močvaru na nekoj vrsti dinosaurusa Ako stisneš pucanje i gore tvoje oružje će napraviti krug od 360 i ubiti neprijatelje koji napadaju iz zraka

Kada završiš ova 4 nivoa dobijaš šifru za drugi dio Ona glasi 18757

Drugi dio je platformska igre Naša junakinja je opremljena mlaznim rancem koji joj omogućava visok i dugačak skok Na površini nemaš oružje nego ti pomaže nepoznati saveznik koji protivnika ubije dodirom U podzemnom lavirintu možeš uzeti pištolj ili dezintegrator Prati mapu Prvo nabavi oružja Zatim pronađi propusnicu (veliko slovo S) i moći ćeš proći kroz vrata sa rešetkama Kada prođeš kroz vrata. idi do kraja lijevo padni dolje i probij se do helikoptera (na desno) Helikopter će te prevesti prema tvom cilju Idi desno do kraja (čuvaj se meteora) skoči na platformu i sa njenog lijevog ruba skoči gore idi dalje lijevo do kraja i Energija je prikazana u obliku srca Svaki dodir sa neprijateljom ti smanjuje srce a na kraju gubiš život

Inače moja je verzija pokvarena tako da većih problema nisam imao A sada Spectrum u ruke i na posao

Igor Pintar



COSA NOSTRA

Za razliku od većine ostalih igara gde ste obično na strani pravde ovde je stvar potpuno suprotna u ulozi ste okorelog mafijaša koji treba da uništi konkurenciju u jednom malom sicilijanskom gradu (verovatno Palermu)!

Pošto ste dugo bili u tuđini u vašem rodnom gradu su drugi preuzeli posao pa ste prinuđeni da se vratite. Tek što ste raspakovali kofere u lokalnom hotelu i izašli da izvidite situaciju nekoliko plaćenih ubica vas je zasulo kišom olova Ali ni vi niste nevni nekoliko hitaca iz vašeg „Magnuma" će ih zauvek ućutkati Na početku imete samo šest metaka ali zalo rezerve municije možete obnavljati tako što ćete skupljati kutije metaka koje ostaju na ulici kada sredite protivnike Morate se navići i na to da ste sami protiv svih jer je i policija protiv vas pošto ste okoreli mafijaš. Pored svih ovih ljutih protivnika može vam se desiti da vam na glavu padne saksija ili da vam neko sa prozora baci bombu pred noge!

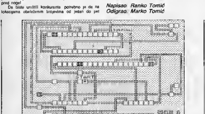
Da biste uništili konkurente potrebno je da na lokacijama obeleženim brojevima od jedan do pet pobijete sve likove koji se na njima pojave (obično po dvojica) U ove lokacije spada i policijska stanica obeležena brojem dva Za svaku od ovih lokacija dobijate čestitku u vidu kratke poruke o uspešno obavljenom poslu propraćene nagradom od 500000 poena

Kada obavite ovih pet akcija igra se završava a ispisuje se neka poruka na italijanskom koju nismo uspeli da pročitamo jer se skoro odmah obriše (ako ste dobar poznavalac italijanskog jezika pokušajte da je protumačite)

Igra se može igrati tastaturom i to ovim tasterima Q—gore A—dole O—levo P—desno SPACE—pucanje Takođe se može koristiti kempston džojstik Ako mislite da Vam je partija loše počela možete je prekinuti pritiskom na CAPS I SPACE istovremeno Puno sreće u stazanju na mafijaški vrh naravno samo u igri!

PS Ako Vam je igra i pored priložene mape previše teška uneske sledeći poke za bezbroj života 39706 183 ili 38841,0 za bezbroj metaka

Napisao Ranko Tomić
Odigrao Marko Tomić



CALIFORNIA GAMES

U uvodnom meniju možete izabrati takmičenje u svim ili samo u jednoj disciplini uvježbavanje ili pregled najboljih rezultata Možete se takmičiti pod okriljem neke od poznatih firmi a u igri učestvuje osam igrača Na raspolaganju su vam šest disciplina Prva disciplina je „half pipe" Sa skejtbordom nalazite se na vrhu polucijevi i cilj vam je da kada dostignete dovoljnu brzinu izvedete jedan od tri što kreativnija skoka Raspolažete sa tri pokušaja i minutom vremena

Druga disciplina je „foot bag" u kojoj pimplujete lopticu pri čemu dobijate dodatne poene ako pogodite pticu koja povremeno prelijeće iznad vas Koristite što više udaraca jer tako dobijate više poena dok za upotrebu jednog ili dva dobijate neznatan broj bodova.

„Surfing" je treća disciplina i cilj vam je da na dasci vozeći se po kalifornijskim valovima dođete do obale Od sudija koji vas ocjenjuje najdarežljiviji je po pravilu pripadnik ljepšeg pola Ovo je jedna od najlakših disciplina „Skating" (klizanje) je usavršeno i nije to više ono jednolično klizanje iz „Zimskih igara" U ulozi ste djevojčica koja nastoji što brže proći cestom koja je puna prepreka Pokatkada joj to ne uspije te svoju ljutnju iskazuje lupanjem nogama i rukama o pločnik U ovoj disciplini nema popravnog ispita

Naslov sljedeće discipline „BMX" ukazuje da je riječ o vožnji biciklom ali ovaj put još luđoj sa još više prepreka Na raspolaganju su vam dva salta oko obje ose po jedan koje koristite pri najvećoj brzini i na najvećim brežuljcima Ako volite veliku brzinu sa puno skokova uključite auto-fire i krenite u avanturu koja će trajati veoma kratko jer je skoro nemoguće kontrolirati bicikl Posljednja disciplina „Flying disc" je po meni i najslabija. Prvo kontrolišite bacača frizbija brzinu i izbačaj a zatim hvatača Nakon toga uloge se mijenjaju i sve tako dok vam frizbi ne padne na zemlju

Said Hrelja

„THROUGH THE TRAPDOOR"

U ovom izvrsnom nastavku tvoj zadatak je da lobanju koja ti je pomagala u prvom delu spasiš iz kandži skelet ptice Novina u ovom delu je to što pored Berka možeš da kontrolišeš i žabu-Druta. Igra je podeljena u četiri dela

ZAMAK. Prvo ćeš videti kako skelet ptica otima lobanju a onda uzmi Druta pa padni dole u kapak koji se otvorio Kao Drut idi levo i skoči pa skini ključ Ovde će ti smetati veliko zeleno čudovište koje leti na tade i ne da ti da skočiš do ključa Postupi ovako idi skroz levo do kraja pa skoči gore tri puta Onda pritisni levo pa skoči gore dva puta i pritisni desno Zeleno čudovište će te sprečiti ali skok je dovoljan da se skine ključ Kao Berk kreni po ključ idi desno pa upadni u otvor na podu Pređi u Druta i idi u sobu gde je ključaonica Skini predmet koji visi gore Samo pazi Drut se kreće u dva ravni koje menjaš pritiskom na dole Sve predmete skidaš odozgo samo se gornje ravni Kada predmet padne odguraj ga u rupu Kao Berk uzmi predmet i pojedi ga (pritiskom na gore) Uzmi ključ pa ponovo pritisni gore i gle čuda Berk može i da leti Zajedno sa ključem odleti desno u sobu sa vratima pa ih otključaj Dovedi i Druta u ovu sobu pa ga kao Berk uzmi i prođi sa njim kroz vrata Eto te u drugom nivou

PODRUM ZAMKA Kao Drut dođi iznad otvora na tavanicu pa skoči nekoliko puta i skinućeš predmet koji je potreban Berku Pređi u Berka pa ga pojedi i on će ti omogućiti da skačeš kao žaba. Sada idi u sobu desno U sledećoj sobi levo su dva kiklopa-patuljka Njih treba preskočiti i to nije naročito teško Ali u drugoj sobi su još dvojica pa kada se sve sklopi dobija se šema prolaženja koja uspeva samo uz mnogo prakse i truda Ovde ću napomenuti činjenicu koja je veoma korisna a to je da Berk može da kontroliše skok. Pazi samo da ne udariš glavom u tavanicu jer ćeš morati sve iz početka Kada i to prođeš dolaziš do soda u kojoj se na sredini nalazi stvar koja puca u vis Sada pređi u Druta pa skini predmet koji visi u toj sobi a kao Berk preskoči pucaljku Uzmi predmet i pojedi ga Sada možeš da uzmeš pucaljku samo pazi da te metak ne pogodi Vrati se desno i pusti da te mali kiklop uhvati Sačekaj da pucaljka ispali metak i on će pasti kiklopu na glavu i ubiti ga Sada to ponovi još tri puta i svi kiklopi će biti mrtvi Uzmi pucaljku i odnesi je u sobu sa ključaonicom Stavi je ispod otvora na tavanici Kada prvi put opali skinuće ključ koji se nalazio gore Uzmi ključ otključaj vrata pa sa Drutom kreni pravo

IZVAN ZAMKA Ovde ćeš odmah na početku naići na teg koji ti nije potreban Ako pokušaš da prođeš treću sobu desno iznenada će se na tade stmeknuti ogromno čudovište i poješće te Zato uzmi predmet koji se nalazi na tom ekranu i idi u sobu levo Stani na sredinu i pojedi ga Berk će poleteti ne gore a ti nećeš moći da ga kontrolišeš Pređi u Druta i brzo ga preko platforme prebaci u sobu u kojoj tvoj prijatelj lebdi u vazduhu Skoči najviše što možeš gore a onda skoči na Berka. Gurnućeš ga tačno toliko da će on posle pada pasti na gornju platformu. Kao Berk idi desno U ovoj sobi nalaze se tri zmijice sa nosevima koji svetlucaju Iznad njih stoje tri kućice Svaku zmijicu treba da staviš u odgovarajuću kuću U slučaju da u ovoj sobi malo zastaneš pojaviće se ogromna kandža koja će te veoma često bacati dole tako da ćeš morati sve ponovo Nju ćeš zavarati tako što kad zastaneš umesto da uzmeš neki predmet ti beži a dok se ona bude podizala gore uzmi predmet U većem broju slučajeva zmijice treba ovako da rasporediš Prva u prvu kuću druga u treću a treća zmijica u drugu kuću Posle toga se neda će pasti predmet ali ti ga ne uzimaj već se vrati u početnu sobu Pretvori se u Druta pa donesi prijatelju predmet koji se nalazi u sobi desno od sode sa kandžom Ovo će ti se možda učiniti teško međutim uz malo vežde sve se može Kao Berk uzmi taj predmet i pojedi ga u sobi gde se nalazi džinovska glava koja te ubija Leti gore pa desno Glave će krenuti za tobom a ti je hladno prateti i upadni u sledeću sobu Ovde ne smeš stati na poklopac koji se nalazi na podu a pošto je ključ iza njega moraš preko Zato ponovi onaj postupak sa Drutom pa prelefi i taj poklopac. Kada budeš došao do ključa ostavi ga tako da ti ga Drut može dogurati (do sada nije mogao) a ti sada možeš i da staneš na poklopac nije važno Pređi u Druta pa doguraj ključ do soda se vratime i onda je lako

ZAMAK SKELET — PTICE Kao Drut idi skroz desno U poslednjoj sobi skini ključ koji visi iznad rupe (možeš i iz rupe) Pređi u Berka pa u drugoj sobi desno idi i pojedi predmet koji se nalazi na podu On ti daje skok. Preskoči zid i došao si u sobu sa ogromnim kosturom Njega za sada ostavi na miru Sada moraš izvesti veliki skok (pazi nema tavanice skok ti je neograničen) u sobu levo Tamo uzmi ključ. Ako padneš u rupu pazi se unutra su zmije koje će munjevito razjapiti svoje čeljusti i krenuti na tebe Međutim one ne ide do samog kraja a posle napada ponovo se na neko vreme povlači u svoju jazbinu Za izlaz iz rupe koristi predmet koji se u njoj nalazi i omogućava ti da letiš Pri tom pazi da ne naletiš na duhe jer ćeš onda morati sve ponovo Sigurno si u ovoj sobi primetio i lobanju koja neverovatno podseća na tvog prijatelja ali ona je tu postavljena samo kao zamka Naime one ničemu ne služi a lobanja koja ti treba nalazi se na sasvim drugom mestu Uzmi predmet koji se nalazi ispod lažne lobanje pojedi ga pa uzmi i ključ Sačekaj da kostur ptica poleti nazad pa poleti i ti na istu stranu Ključ ostavi pored vrata Uzmi lop koji se nalazi u toj sobi i napuni ga kapljicama koje padaju sa krova tako što ćeš stati ispod mesta gde prokišnjava zajedno sa topom Top ostavi na sredini soda a u ulozi Druta doguraj ga do samog kraja Doskoči do kostura i približi mu se do maksimuma On će napraviti korak i krenuti ka tebi To ponovi nekoliko puta dok duša koje leti (iz topa) ne pogodi borbeno raspoloženog kostura Od kostura će posle ovoga ostati samo lobanja koju je čuvao a po nju si i došao Donesi je u sobu sa vratima Otključaj ih uzmi lobanju i prođi kroz vrata

Naravno da daz pouke ne možeš ni pomišljati da završiš igru. On se ubacuje ovako kada se učitaju tri dela igre slika, i dva duža mašinska dela brejkuj program pa otkucaj POKE 47492 0:LET=USR 34200. Srećno igranje

Vladimir Janković

INDOOR SPORTS

1) AIR HOCKEY

Kada učitate ovu igru prvo će se pojaviti meni koji vam dozvoljava da birate broj utakmica potrebnih za konačnu pobedu (1—4) pita vas da li želite promenu strane a možete odabrati i nivo (beginner normal ili hyperdrive). Ukoliko se odlučite za igru protiv kompjutera, možete odabrati i težinu kojom će on igrati protiv vas (easy average hard ili proffesional).

Scenario i cilj ove igre su vrlo jednostavni. Nalazite se na hokejaškom igralištu a svaki od igrača ima pljosnatu pločicu koja na vrhu ima i nekakvu ručicu. Na krajevima igrališta nalaze se otvori. Odbijanjem peka od rubova igrališta potrebno je ubaciti ga u protivničku rupu. Igra se do 12 postignutih golova. Onaj tko se duže vreme druži sa Spektrumom setiti će se da je ova disciplina veoma slična igri XENO koja je ranije napravljena za ovaj računar.

2) DARTS

U meniju birate broj igara (1—9) broj poena od kojih će igra startati (301, 501 ili 701). Možete se odlučiti za istovremeno takmičenje sa protivnikom ili za uvežbavanje pojedinih poteza (opcija SOLOPLAY). Posebnu pažnju obratite na dve stvari. U donjem desnom uglu nalazi se natpis ANGLE (označava kut pod kojim će leteti pikado strelica). Najbolje je da to podesite između 45 i 47 stepeni. Određeni kut dobivate na taj način da strelicu najpre pomerite do željenog položaja na meti a zatim stisnete pucanje. Počet će se vrteti stepeni i kada naidete na željeni kut otpustite pucanje. Tada će se u levom donjem uglu pokazati tri pokretne crte koje označavaju visinu na kojoj će strelica leteti. Ispočetka je „uštopajte" kada se pojave dve i pol crte. Kada to učinite dvodimenzionalan prikaz igre se briše a pokazuje se vaš (ili protivnički) igrač u trodimenzionalnoj prostoriji i vidite tek ispaljene strelice.

3) PING—PONG

Prvi meni je gotovo istovetan onome kod AIR-HOCKEY-a. Razlikuje se samo u tome što poseduje opciju PADDLE CONTROL (pomoću nje određujete da li ćete sami upravljati reketom ili će kompjuter to činiti umesto vas — MANUAL MOVE ILI AUTO-MOVE). Drugi meni pokazuje jačinu vaših i protivnikovih pojedinih udaraca (smashing forehand backehand i sl.) te ukupnu vašu i protivnikovu vrednost.

Igra se do 21 poen ali postoji mogućnost da i ranije pobedite. Ukoliko postignete 11 poena a vaš protivnik niti jedan istovremeno se završava set u vašu korist (kod nas se ovo popularno zove „automat"). Morate biti veoma pažljivi jer je program jako osetljiv na komande. Tako na primer ako želite servirati povucite palicu vrlo malo nagore i stisnite pucanje. Isto tako učinite i kod „rezanih" lopti (jedino još dodajte željeni smer). Evo još dva saveta. U glavnom meniju uzmite PADDLE CONTROL pa nećete morati da se mučite sa pomeranjem reketa već ćete voditi brigu samo o udarcima. Isto tako odredite da između setova nema zamene strana nego uzmite onu gde gledate lice reketa. Biće vam mnogo lakše.

4) TP BOWLING

Uz već standardne opcije u meniju ovde birate još i broj prvih i drugih kugli. Ovo mi je najmilija disciplina u celom programu. Lepo animiran kuglač i pregledna obrada rezultata koje vidite u dnu ekrana učinile su ovu disciplinu vrlo uzbudljivom za igranje. Rezultati svakog bacanja označeni su u jednom od kvadrata i to na sledeći način. U dnu kvadrata označen je konačan broj poena iz 2 bacanja, samo rezultati iz „čišćenja" (ili drugog bacanja) a odmah kraj njega rezultati prvog bacanja. I šta reći u završnom komentaru. Ukoliko se volite takmičiti sa prijateljima, a da pri tome ne morate „prosipati" nerve nabavite ovaj program sigurno se nećete pokajati.

Zlatan Hamzić

Aliens 2

Ako ste gledali film „osmi putnik 2" u igri nećete imati neke veće probleme. Igra je u stvari ista kao i film sa nekim manjim izmjenama. Izabrani ste da predvodite skupu specijalno obučenih S.F. komandosa na udaljenu planetu Alfa 5. Na toj planeti su se doselili ljudi i izgradili malo svemirsko naselje sa nuklearnim reaktorom. Međutim iz utrobe planete su izašli vanzemaljci nastanili se u ljudskim utrobama i namnožili se u velikom broju. Vi stižete prekasno. Preživjelih više nema. Ostaje vam samo borba prsa u prsa sa vanzemaljcima. Vaš zadatak je da ih ubijate što je moguće više. Imate sedam članova posade. Možete voditi svakog od njih ako na tastaturi pritisnete prvo slovo njegovog imena. Kad vam se lik pojavi s desne strane vidite crtu u raznim bojama to su naboji za vaš mitraljez (100 metaka ograničenje). Kada ih sve potrošite nema vam spasa. S lijeve strane vidite kako vaše srce radi. Kad crta postane ravna oprostili ste se sa životom. Od silnog trčanje po vasionskom naselju vaš nišan će se kretati sve sporije. Malo se odmorite pa nastavite dalje. Umorni ćete vidjeti po žutoj ili zelenoj crti ispod vašeg imena. Što se crta više smanjuje više ste umorniji. Ukupan broj sobe kroz koje morate proći je 248. Kad ste blizu skrepa koji hoće da bude podstanar u vašoj utrobi čuje se zvuk koji postaje sve jači. Kad ugledate skrepa kako se bezazleno kreće naniśanite mu u glavu opalite (potreban je najčešće jedan metak) i on pada kao pokošen. Ako ne opalite na vrijeme kad vas skrap ugleda kreniće velikom brzinom prema vama. Tada mu luđačkom brzinom opalite 5—6 metaka pored glave dok mu se ona ne razdvoji na dve polutke. Ako ne uspijete on otvara svoje strašne vilice i ... imate podstanara koji ne plaća mjesečnu kiriju. Na zidovima vidite jaja od vanzemaljaca u obliku vinove loze. Njih jednostavno napucavate dok ne nestanu. Na zidovima ćete vidjeti i vrata kroz koja prolazite tako što navedete nišan na njih pritisnete razmaknicu i već ste u drugoj prostoriji. Vrata koja nemaju terminal (mali kvadratić ispunjen narandžastim i bijelim kvadratićima) ne možete proći. Kada ubijete kraljicu spasili ste se i uništili nepoznatu vrstu organizama iz svemirskih bespuća.

Ukoliko vam nešto nije jasno nazovite na telefon (055) 237-631

Ivan Jakobović
& Damir Salantić

Rapids

Rapids (na engleskom — brzak) je simulacija vožnje kanua na brzim vodama. Imate mogućnost da igrate sami protiv prijatelja, da napravite novu stazu ili da pročitate instrukcije. Staza ima četiri i poređane su po težini. Za početak odaberite stazu pod brojem 1. Cilj je proći kroz sve kaplje u što kraćem vremenu. Promašaj kaplje povisiće kazneno vreme od 5 sekundi.

Najveći deo ekrana zauzima staza gledana iz „ptičje perspektive". U desnom uglu nalazi se uvećan takmičar i tu možete da pratite sve pokrete koje napravite za vreme trke. Tu je naravno i vreme koje je proteklo kazneno vreme najbolje vreme napravljeno na stazi koju vozite i vaša energija.

Najzanimljivije opcije je sigurno pravljenje nove staze. Prvo pravite novu reku (edit river). To radite tako što palicu vučete prevo gore ili dole u zavisnosti od toga kako želite da izgleda reka. Zatim upisujete brzinu reke na pojedinim delovima. Brzina koju možete da upišete kreće se od 1 do 16. Najpametnije je najveću brzinu reke upisati na pravim delovima a nešto sporiju na krivinama. Posle toga postavljate stene ostrva i nasipe (edit river features) brzake (edit rapiden) i kaplje (edit courses). Svoju stazu možete da snimite ili odmah da je isprobate.

Srećno!

Rastko Popović

AIR RALLY

Na auto relije smo navikli a ovo je prilika da se okušate kao pilot sportskog aviona. Komande su jednostavne (gore-dole levo-desno) problem je prevesti određenu etapu u zadatom vremenu a nužno je i spretno upravljanje avionom.

Ako letite nisko prepreke su mnogobrojne, ako se podignete više trebaće vam mnogo umeća da zaobiđete letilice koje vam dolaze u susret a od ružnih stvari koje vas mogu zadesiti kulminacija je ispadanje iz aviona posle koga ćete se prizemiti nasred Svinjca!

Jedini savet koji mogu da vam dam da se strogo držite pravca i uspećete.

Dejan Dojčinović

THUNDERCATS

Zli Mum-re je ukrao Oko Grome. Mladi Lion-o (to si ti) poslat je sa nebe da krene u potragu za kradljivcem. Međutim Mum-ra šalje na našeg junaka legije pakla koje trebaju da ga zaustave.

Dok čudotvornim mačem nalik na Konana krčiš put ka izlazu nivoa, pred tobom se pojavljuju neprijatelji najrazličitijih oblika. To su šarane lopte ptice ljudi nosorozi (?) patuljci i drugi. Da bi izbegao bliski kontakt s njima osim mača koristiš saginjanje i skokove. Posle svakog nivoa računar ti odbrojava bonus poene za broj ubijenih neprijatelja i vreme sa koje si stigo do kraja. Izlaz iz nivoa ćeš prepoznati po tigrovoj glavi. Inače budi svestan da si sve vreme pod nadzorom boga Oluje koji u jednom trenutku može da se naljuti i baci na tebe grom koji trenutno ubija. To se u igri retko dešava ali u trenutku kada je Olujni bog ljut na tebe možeš izgubiti jedan od šest života koliko imaš na početku igre.

Dok se krećeš po ravnicama i pećinama sa drveća i stena skidaš sanduke (nazovimo ih tako) u kojima ćeš naći razne predmete nekad preistorijsku pucaljku nekad nagradni život a nekad ništa. Ako skineš pucaljku koja je veoma korisna u daljoj borbi, nemoj reskirati dirajući druge sanduke jer može da se desi da ti se ponovo vrati mač.

Tokom cele igre strelica u donjem delu ekrana ti pokazuje pravac kojim treba da ideš. Posle prva tri nivoa odabereš jedan od četiri puta kojim ćeš ići dalje. To su Vatra Voda Nebo Zemlja. Redosled nije bitan ipak krenimo redom.

VATRA. Krećeš se udesno. Lopte koje dolaze sa svih strana su veoma opasne. Ne možeš ih udariti mačem već da bi ih izbegao koristiš skokove i savijanje. Posle nekog vremena naići ćeš na zalutalo vozilo sa kojim ćeš lako preći ovaj nivo.

VODA. Imaš pucaljku i ideš udesno. Napadaju te dosadne ptice i debeli nosorozi koji se kreću u skokovima. Sve ih tamaniš mitraljeskom paljbom. Mnogo veći problem je živo blato zbog kojega treba biti izuzetno precizan pri skokovima.

NEBO. Opet si sa mačem. Osim poznatih živuljki javljaju se i neke nove manje opasne lopte. Pucaljku i nagradni život tražiš ispod pečurki i četvorouglove koji se nalaze na zemlji. Nemoj skakati na njih jer ćeš izgubiti jedan život već ih najpre udari mačem. Ovo je jedan od najtežih nivoa.

ZEMLJA. Ovde se pojavljuje mistična statua koja baca na tebe vatrene kugle. Treba joj što brže prići i čučnuti. Da bi nestala potrebno je dosta udaraca. Osim vatrenih kugli čuvaj se neprijatelja koji prilaze sa leđa.

Nakon još nekoliko nivoa koji su međusobno dosta slični računar te obaveštava o poslednjoj finalnoj bici sa snagama opakog Mum-re. Strelica pokazuje ulevo ali ti pođi suprotnim smerom. Posle duge borbe ponovo ćeš naići na vozilo. Uđi u njega i idi ulevo.

Tu je i kraj ove sjajne igre.

Jovan Zečević

DRAUGHTS GENIUS

Ova igra po Jugoslaviji kruži pod imenom DAMA. Igra se po engleskim pravilima pa protivnika morate jesti kad se nađete u odgovarajućoj poziciji. Možete da igrate protiv kompjutera protiv prijatelja ili gledati kako kompjuter igra sam protiv sebe. Kompjuter je dobio lik Ajnštajna u ovoj igri. Veliki profesor je zaista vrlo lepo nacrtan. Sa vama komunicira preko oblačića kao u stripovima. Kad razmišlja češka se po glavi a ako predugo razmišljate zevanjem će staviti do znanja da mu je dosadno a ako baš preterate-zaspaće. Kada pobedi veselo će dići ruke u vazduh a ako ga pobedite pljeskaće vam kao i svaki pravi sportista. Tabla je krupno nacrtana u tri dimenzije. Figure pomerate tako što stanete na odabrano polje i pritisnete pucanje a zatim pređete na polje koje hoćete i dvaput pritisnete pucanje. Kraj table se nalazi nekoliko veoma korisnih ikona. One znače sljedeće: R-Ajnštajn će odigrati umješto vas ako u tom trenutku ne možete smisliti nijedan pogodan potez.

Strelice za gore/dole — povratak u prvobitnu situaciju. O — završavanje igre.

Žiža Gordan

396

AGENT X 2
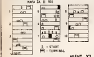
Igra se sastoji od tri dela. Kad završite jedan dobijete šifru za sledeći.

Prvi deo

Nebo iznad grada je prepuno svakojakih neprijateljskih eskadrila (raznih boja i formata). Na vama je da ih sredite. Oružje vam je u početku skromno, ali kada skinete čitavu eskadrilu kao nagrada na ekranu se pojavljuje limun. Kada ih sakupite ukupno četiri — dobijete laserski pištolj (njegovo dejstvo ne traje dugo) ali ako ste brzi pa u svoju limunsku kolekciju dodate još četiri komada, onda ste dobili i jagodu pa možete da iz lasera pucate od mile volje. Na kraju vas čeka zelena neman pa kad i nju odrakate evo šifre za drugi nivo HERE COME OL FLAT TOP

Drugi deo

U kompjuterskom centru morate na terminalima ukucavati određeni kod koji svetli na ekranu. Soba ima deset i u svakoj je po jedan terminal. I naravno da je stvar u redosledu. Kada vidite na ekranu poruku ENTER CODE, pritisnete taster za gore (UP) i dugme za pucanje (FIRE). Na kraju opet vas čeka neprijatelj. Sređujete ga uz dosta pucnjave i dobijete šifru za treći nivo THERE S NO ESCAPING IT



Treći deo

Pomoću palice i lopte uništavate cigle. Kada napravite otvor, preciznim udarcem šaljite loptu u kraljevsku ložu.

I onda je sve gotovo.

Zoran Mirčevski

MEAN STRIKE



Na motociklu jurite po stazi, a život vam zagorčava protivnik koji zamislite puca na vas. Razume se, vraćate mu istom merom, gađajte ga mitraljezom, ispaljujte rakete na njega, gurate ga sa staze, ispuštate ulje.

I tako do cilja, ako vam to uopšte pođe za rukom.

Usput sakupljajte ulje (buriči sa oznakom OIL), benzin (buriči sa oznakom FUEL) i dodatne rakete koje liče na šipke pored staze.

U donjem delu ekrana su razni pokazivači — količina goriva i ulje, broj raketa, merač vremena, broj života i radar.

Komande su:

Pucanje — mitraljez

Dole plus pucanje — skok

Gore plus pucanje — raketa

Levo (desno) plus dole plus pucanje — prolivanje ulje

Kome se dopada neka izvoli!

Boris Mladenović

OUT RUN

Pred sobom imamo jedan odličan program, koji simulira vožnju automobila (podseća na „ferari") na širom auto-putu, po celom svetu. Sama igra ima pet delova a svaki deo ima pet nivoa. Delovi su: Vine yard, death valley, desola Hill, autobanh i lakeside. Pre same vožnje biramo muziku na radiju u automobilu. Imamo dve odlične urađene melodije i mogućnost isključenja radija. U najvećem delu ekrana odvija se igra, dok u gornjem delu vidimo: vreme za koje treba preći stazu, rezultat, vreme kruga (nivoa), a u donjem brzinu (imamo dve brzine, najveća je 294 kilometra na čas, međutim meni nije smetalo da idem i 490 kilometara na čas. To se dešava kada se posle udarca blokira brzina, pa krećete od 294 kilometara na sat) i nivo koji vozite. Prepreke pored puta su: palme, žbunje, kuće, kamenja... i svaki udarac pri velikim brzinama uzrokuje prevrtanje automobila i ispadanje iz njega što je lepo grafički urađeno.

Na putu vam smetaju i druga vozila (kombiji, automobili...) koji su grafički bolje urađeni nego u „test drive"-u. U ovoj igri glavna vam je prepreka vreme koje naumitno ističe. Udarac u druga vozila vam samo smanjuje brzinu, što se opet plaće vremenom. Komande su: gore—gas, dole—kočnice, dugme—menjanje brzina. Na kraju ne dobijate nagradu u vidu male lepe muzike (vidi p.p.) već jedan sočan poljubac od svoje partnerke koja veselo tapše, a vi se držite ponoseno. Međutim, partnerka celim što vas nagradi na kraju dela je vrlo nezgodna pri svakoj grešci prilikom vožnje. Šta još reći o igri? Ukratko — SUPER. Pored „test drive-a" najbolji simulator vožnje na C 64.

Zorić Vladimir

SVEŽE UČITANO

Komodor

TRAZ

TRAS



Da je na Zemlji, zvao bi se Alkatras, ali pošto je negde bestraga u svemiru, prva su mu dva sloga odsečena.

Ucstalom, sve je u imenu, što rekla Đulijeta. Suština je nepromenjena. I Tras je zatvor. Potpuno je kompjuterizovan, o bekstvu nema govora... ali o čuda, kompjuter se pokvario i vi pokušavate da se probijete na slobodu. Rušite zidove, ciglu po ciglu...

... i shvatate da ste ovu igru i ranije igrali, samo što se tada zvala drukčije. Recimo, ARKANOID.

Ali ne očajavajte. Ima novina. Najveća je u tome što imate dva maška. Ona druga se kreće gore-dole — i, samim tim, čini igru živahnijom.

Uz to, ako niste zadovoljni zidovima koje je smislio kompjuter, možete praviti druge po svom nahođenju. Pomislite samo, projektovati svoj sopstveni zatvor! Šta čoveku više treba da bi bio srećan?

I BALL 2

JA, KUGLA 2

Kugla kao i svaka druga? Ne. Ova je inteligentna i uz to se amaterski bavi arheologijom. Ona skakuće po podzemnim lavirintima iz vremena pradavnih, nastojeći da istraži prošlost svoje rase. Pedeset lavirinata sve u svemu, na kraju svakog pelog čeka neki dragoceni i naučno korisni predmet iz starine.



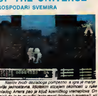
Ali kako stići do njega? Treba naći ključ da bi se iz jednog lavirinta prešlo u drugi, treba gurati i rušiti stene, treba se čuvati letećih radioaktivnih gromada, treba se boriti sa mutantima, svašta treba.

A kugla ima samo neku vrstu bacača plamena i to joj je sve oružje. Doduše, nađe se usput i pokoji energetski kamen (čuj, energetski kamen!) koji joj daje veću brzinu ili jači odskok ili ubojniju vatrenu moć ili neku drugu prednost. Ima i „pametnih bombi" sakrivenih pod kamenjem.

Razgrćući to kamenje (i tako katkad nalazeći korisne predmete i skrivene izlaze) kugla će možda stići do kraja, a možda i neće, ali u svakom slučaju steći će mišiće kao kamen.

MASTERS OF THE UNIVERSE

GOSPODARI SVEMIRA



Naslov zvuči dozlaboga pompezno, a igra je manje-više jednostavna. Idiotskim slicajem okolnosti u ruke mladog Amera pao je ključ kosmičkog vremeplova. On misli da je to muzički instrument (bistrog li momka!), ali vi znate pravu istinu i krećete u akciju.

Na žalost, kreće i vaš protivnik, zli Skeletor sa gomilom svojih sledbenika. Morate se žestoko boriti s tom bandom dok tražite osam akorda (jer samo oni aktiviraju ključ).

Ako preživite borbu na otpadu, u prodavnici elektronskih uređaja i na krovovima kuća (i ako ste pokupili sva akorde) sukobljavate se sa Skeletorom koga morate ugurati u jamu i tako, valjda, uništiti zanavek.

Na otpadu se tučete koristeći noge i ruke, a u prodavnici i na krovovima pucate. Na putu od jednog poprišta do drugog borite se sa smetalima koliko da vam ne bude dosadno.

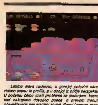
Kad i ako pobedite, čeka vas malo razočaranje: nećete postati gospodar svemira.

OCTAPOLIS

Oktapolis je planeta sa (naravno) osam utvrđenih gradova. I jedino nju Galaktička imperija nije uspela da...

No to možemo mirno zaboraviti. Imamo posla sa igrom strogo pucačkom. Prvo letimo iznad planete i gađamo sve što nam se ispreči, a posle se spuštamo na površinu planete, ulazimo u grad i... i takođe gađamo sve što nam se ispreči.

Onda opet sve ispočetka, dok ne pređemo svih osam gradova.



Letimo sleva nadesno, u gornjoj polovini ekrana vidimo scenu iz profila, a u donjoj iz ptičje perspektive. U početku ćemo imati probleme sa sletanjem kasnije, kad nakupimo dovoljno poena o pravom trenutku obaveštavaće nas alarmni signal. Poeni (mnogo mnogo poena) daju i nove živote.

Jednom reči ubilo se za ljubitelje pucanja koji bi hteli da vide neku novu scenografiju i neka nova smetala.

Spektrum

SALAMANDER



Bivali smo već u toj situaciji igrajući TRANSMUTER i NEMESIS. Leteli smo kroz mračne podzemne hodnike načičkane pokretnim i nepokretnim smetalima, nastojeći da se provučemo do kraja za koji smo znali da mora postojati, ali koji nam je večno izmicao.

Tako je i ovde s tim što postoji mala predigra. Prvo skidamo neke neidentifikovane leteće predmete koji nailaze sve četiri po četiri. Kako koju eskadrilu likvidiramo, tako stičemo poneki dodatak. Vrlo je lako iz predigre izaći sa tri laserska mitraljeza i topom koji puca u tri pravca istovremeno.

U igri nam prete topovi meteoriti veliki zubi itd., uključujući predmete koji liče na ogromne paučje šape. One su ranjive samo na jednom mestu, članku koji treperi. Upucamo ga i cela šapa nestaje. Ostale uništive prepreke dovoljno je pogoditi bilo gde (Naravno ima ih na tone, dušu dalo za mazohiste).

GARFIELD

GARFILD



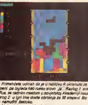
Evo njega! Sklon jelu spavanju i dokonim mislima sa očima uvek na pola dvanaest dobroćudni i dragi (i debeli) mačor Garfild izlazi iz stripa i ulazi u naš kompjuterski život.

I odmah ga muči krupna muka. Njegova nekadašnja draga pala je u šinterske ruke i valja je osloboditi. Garfild kreće u potragu, skuplja korisne predmete (neke mora otimati od svog drugara psa Odija) savlađuje prepreke na ulicama i ispod njih, preganja se s pacovima i uličnim palma I jede sve što stigne. Jelo mu je vrlo bitno u ovoj igri ako ga ne budete redovno hranili nećete daleko doterati.

Isto tako, morate ga puštati da se odmara. Ovaj podatak može delovati ponešto čudno ako znamo da kompjuterske igre traže brzinu i kažnjavaju oklevanje, ali moramo ga pojmiti i po njemu se ravnati. Ergo kad se Garfild zamori zaslanite u igri dok ne odrema jednu turu. Biće mu lakše da nastavi potragu.

Dobro je, uostalom podsetiti se katkad da nije sva sreća u sumanutoj jurnjavi.

TETRIS



Primetićete odmah da je u naslovu R okrenuto za 90 stepeni pa izgleda kao rusko slovo „Ja". Razlog 1 autor je Rus, sa radnim mestom u sovjetskoj Akademiji nauka. Razlog 2 u igri ima dosta obrtanja za 90 stepeni što će vas namučiti žestoko.

Muke dolaze otud što je igra krajnje prosta. Sa vrha ekrana padaju razni geometrijski oblici. Vi ih morate slagati kao zidar cigle red po red odozdo naviše i to tako da savršeno „pasuju" tj. da između njih ne bude ni trunke praznog prostora.

Naravno oblici se bitno razlikuju od cigala. Neki su od njih kvadrati neki pravougaonici neki nalik na L ili T ukratko, treba dobro (i brzo!) promozgati pre no što im odredimo mesto. Donekle nam je lakše utoliko što ih možemo obrtati kako nam je volja.

Kad popunimo (bez praznina) i poslednji red igra je završena. To može potrajati — štaviše potrajaće — jednu omanju večnost.

TERRAMEX



Loše vreme. Kec iz istorije. Infekcija. Jednom reči ne možete glavu dići od muka i nevolja kad povrh svega sto još i vesti da se ka Zemlji stuštio asteroid koji će je raznetí u vrlo sitne komade. Jedini je spas (a) naći Ajnštajna (b) naći delova mašine za odbijanje asteroida i (c) staviti mašinu u pogon.

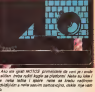
Dovoljno da čovek osedi preko noć. Srećom imate pred sobom petoricu pustolova iz raznih zemalja birate koga hoćete i šaljete ga na posao.

Tačnije u pustinju gde sve vrvi od lešinara i zmija, a kasnije i na druga mesta. Isprva će mu biti čupavo dok ne nađe usisivač ili frulu ili već tako nešto. Pomoću usisivača leti kao kakva savremenija veštica a frulom začarava zmije kako poskoke tako i šarke.

Ima tu još korisnih predmeta (među njih spadaju i delovi mašine) jedino što retko kome možete na prvi pogled pogoditi borbene osobine i namenu. Ako vas bude mrzelo da odgonetate samo sačekajte da asteroid udari u Zemlju tada ćete se osloboditi svih problema.

MADBALLS

LUDE GRUDE



Ako ste igrali MOTOS primetićete da vam je i ovde cilj sličan treba rušiti kugle sa platformi. Neke su lake i brze neke teške i spore neke se kreću načinom predvidljivim a neke sasvim samosvojno, dakle nije vam lako.

Tim pre što ste i sami kugla pa vam se lako može desiti da i sami sletite sa platforme i izgubite jedan život.

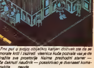
No čim oborite jednu kuglu (i time je zarobite), možete se ako hoćete pretvoriti u nju i steći njene osobine. To radite veoma jednostavno padanjem u praznu kantu za otpatke.

Dok se muvate od platforme do platforme nailazite na trambuline katapulte rampe i piramide odmah shvatate da vam ti predmeti uglavnom pomažu u životu i redu — ali da vam mogu i odmoći ako niste vešti.

Nailazićete i na Birokrate (Nije vic programer ih je zaista tako nazvao). Kao i u pravom životu velika su smetala. U igri ih valja ubijati — time uostalom dobijate i mnogo poena — dok je u stvarnosti preporučljivo pribegavati manje radikalnim merama.

INSIDE OUTING

PRETRAGA BESTRAGA



Prvi put u svojoj obijačkoj karijeri doživeli ste da se ne morate kriti i zazirati vlasnica kuće pozvala vas je da pretražite sve prostorije. Naime prethodni stanar — inače čaknuti naučnik — posakrivao je dvanaest komada nakita negde.

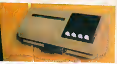
Krenete vi na posao srećni i presrećni — a onda se ispostavi da je kuća puna mutanata mahom ogromnih domaćih životinja (Čaknuti naučnik bio je vredan). Dok vi preturate po kući — a imate i šta svaki je predmet pomičan — mutanti se bacaju na vas kao patka na jelenka. Negde će vam stvari odmah biti jasne (na primer pacova ćete umiriti komadima sira) a negde valja itekako mućnuti glavom.

Naročito kad se radi o guranju dizanju i kombinovanju nameštaja i svakojakih drugih predmeta. No ako budete revnosno istraživali i pomicali sve na šta naiđete možete se nadati da ćete obići celu kuću i naći nakit.

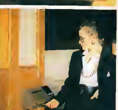
U stotom ili sto prvom pokušaju.

VAŠE RADNO VREME JE DRAGOCENO

NE TROŠITE GA SABIRANJEM ČASOVA NA ŽIGOSNIM KARTICAMA







N Odseku za računarstvo i informatiku INSTITUTA JOŽEF STEFAN razvili smo savremeni sistem za registraciju i obračun radnog vremena koji omogućava.
- umesto žigosnih kartica, magnetne kartice,
- umesto satova za žigosanje, mrežu elektronskih stanica za registraciju,
- umesto »ručnog« sabiranja minuta, permanentan obračun radnog vremena i uz uređenih ispisa.

Zašto je ovaj sistem interesantan za vas? Zato što je tehnička novost? Ne. Zato što je sistem žigosnih kartica toliko skup da ćemo ga sve teže nabavljati. Da li je skup zbog visoke cene uređaja? Ne. Zbog izgubljenih časova kod računanja podataka na karticama.

Zato prepustite računanje računaru!

Postupak registracije je jednostavan: kod dolaska i odlaska povučemo magnetnu karticu kroz zarez u stanicu i pritisnemo na tipku. Na sličan način registrujemo prekovremeni rad, službenu i bolesničku otsutnost, odmor.

Mrežu stanica za registraciju možete da priključite na računar. Za niz različitih tipova računara pripremili smo paket programa koji će vam omogućiti (uz ovlašćenje!) pregled i uređen ispis obračunatih podataka. Kod svakog radnika uzeće u obzir fiksno ili klizeće radno vreme, smene, subote, nedelje i praznike, a u stanice će emitovati kratke poruke (na pr. RADNIČKI SAVET U 15,30)

Primer ispisov

Institut Jožef Stefan — EV-4 Ispis po simbolih — Stran 1

Ispis za čas
Od: 1. Sep. 86
Do: 1. Okt. 86

Org. enota II. 33 — Datum obdelave 20. Nov. 86

Matič. štev.	Priimek, Ime	Del. Obv.	Ure dela	Nad. ure	Služb. izhod	Služb. potov	Oprav. odsot.	Boln.	Redni dopust	Pl. dopust	Nepl. dopust	Oprav. izhod	Priv. izhod	Vsot. salda
398-a	Bartol Anton	195:30	201:42	–	47:30	30:50	8:30	–	–	8:30	–		0:26	0:00
	Bobnar Jelka	198:30	195:42		10:08	–		8:24	–	–		–	–	–
96	Brlek Anton	198:30	206:48	–	9:16	110:30		8:30	–	–		–	–	16:00
	Bačnel Bojan	195:30	195:06	–	61:52	–	76:30	–	8:30	–	–	–	–	
	Černič Jože	195:30	192:43	–	26:13	17:00	–	–			8:30	–		5:00
	Dulić Jugoslav	198:00	171:24	–	31:44	42:30	8:30	3:07		–	–		4:04	–
95	Gruden Marjan	195:30	193:16	–	35:12	17:00	–	–		–	–		4:00	10:00
	Jagodič Janez	195:30	192:14		24:38	57:54	–		51:00	–			–	3:00
	Kalus Ivo	193:30	193:51	–	28:06	93:30	–	–	–	–	–	–	–	11:00
	Lobe Mojca	198:30	182:28	–	16:01	17:00	–	–			–			3:00
M178	Pecnik Bojan	195:30	192:06	–	44:08	82:41	–				–			5:00
	Pihler Bruno	195:30	194:37	–	37:20	42:30	–		78:30	–	–			3:00
	Rozmanec Frančiška	195:30	197:44	–	2:11	–		51:00	51:00	–			–	–
	Semolič Nada	196:30	204:56	–	37:01	–		12:30	34:00	–	–	–	–	–
12/1	Srnc Franc	196:30	197:01	–	47:06	67:14	–	–	–	–			–	6:30
	Urbančič Franc	196:30	196:24	–	26:06	17:00	8:30		17:00			–	–	2:00
	Žibert Danica	196:30	208:42					–		–	–	–	–	–

Skupaj za OE II.	Del. obv.	Ure dela	Nad. ure	Služb. izhodi	Služb. potov	Oprav. odsot.	Boln.	Redni dopust	Pl. dopust	Nepl. dopust	Oprav. izhod	Priv. izhodi
33	3323:30	3308:43	–	473:27	845:39	100:00	84:00	338:00	8:30	8:30		8:30

NOVA FUNKCIJA: evidentiranje i obračun potrošnje toplih obroka u pogonima ishrane

Programski paket daje izveštaje po organizacionim jedinicama i izbirni izveštaj za čitavu organizaciju. Razvrstava se po
- azbučnom redu prezimena ili
- broju kartice ili matičnom broju

Vrste pismenih izveštaja
1. Ispis zbivanja
2. Ispis salda
3. Ispis prekršilaca
4. Ispis po simbolima
5. Ispis prisustvovanja
6. Ispis ličnih podataka
7. Ispis broja prisutnih

univerza e. kardelja
institut "jožef stefan" ljubljana, jugoslavija
Odsek za računalništvo in informatiko
61111 Ljubljana, Jamova 39/p.p. (P.O.B.) 53
☎ (061) 214-399/Telegraf: JOSTIN Ljubljana/Telex: 31-296 YU JOSTIN

REFERENTNI SPISAK — Mart 1988

dosadašnjih instalacija sistema za registraciju i obračun radnog vremena.

Radna organizacija	broj zaposl.	glavni računar
1. SLOVENIJALES DO Trgovina, Ljubljana	1200	IBM 4341
2. ISKRA ELEKTROOPTIKA Ljubljana	1200	DEC VAX-11/850
3. MURA, Murska Sobota	6000	IBM
4. KONUS SI. Konjice	3000	IBM
5. RADE KONČAR, Raz. Institut, Zagreb	1200	ISKRA DELTA 340
6. SMELT Ljubljana	300	IBM PC/XT
7. PROJEKT Nova Gorica	100	ISKRA DELTA - PARTNER
8. TEHNOIMPEX, Ljubljana	100	IBM PC/XT
9. UNIS Savlje, Ljubljana	500	DEC-MICROVAX II
10. BETI Metlika	1200	DEC-MICROVAX II
11. ISKRA DELTA – Ljubljana	1000	ISKRA DELTA 800
12. ISKRA DELTA – Nova Gorica	100	ISKRA DELTA 800
13. SOb Ljubljana Bežigrad	200	IBM PC/XT
14. SOb Ljubljana Moste-Polje	200	IBM PC/XT
15. Raziskovalna skupnost SRS	50	DEC-VAX
16. ELEKTROTEHNA DO ELZAS	200	SCHNEIDER PC
17. ENERGOPROJEKT – Beograd	200	IBM PC/XT
18. LB – Kranj	200	DEC-MICROVAX II

Sistemi u postupku isporuke:
BANEX Zagreb, Elektrokontakt Zlatar Bistrica, Ina Nafta-plin Lendava